

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО**

Факультет інформатики та обчислювальної техніки  
(назва факультету, інституту)

Кафедра автоматизованих систем обробки інформації і управління  
(назва кафедри)

"На правах рукопису"  
УДК 519.854.2

«До захисту допущено»  
В.о. завідувача кафедри

О.А.Павлов  
(підпис) (ініціали, прізвище)  
“ ” 20 19 р.

**МАГІСТЕРСЬКА ДИСЕРТАЦІЯ  
на здобуття ступеня магістра**

за спеціальністю 126 Інформаційні системи та технології  
(код та назва спеціальності)

ОПП Інформаційні управляючі системи та технології  
(код та назва спеціалізації)

на тему: СКЛАДАННЯ ЕНЕРГЕТИЧНО ЕФЕКТИВНИХ РОЗКЛАДІВ В  
ОПЕРАТИВНО – КАЛЕНДАРНОМУ ПЛАНУВАННІ

Виконав: студент VI курсу групи ІС-81мп  
(шифр групи)

Клименко Вадим Михайлович  
(прізвище, ім'я, по батькові) (підпис)

**Науковий керівник** доц., к.т.н. Сперкач М.О.  
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

**Консультант** проф., д.т.н. Томашевський В.М.  
(науковий ступінь, вчене звання, прізвище, ініціали) (підпис)

**Рецензент** доц.каф.АУТС, к.т.н.,доц.Писаренко А.В.  
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає  
запозичень з праць інших авторів без відповідних  
посилань.

Студент  
(підпис)

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки  
(повна назва)

Кафедра автоматизованих систем обробки інформації та управління  
(повна назва)

Рівень вищої освіти другий (магістерський) за освітньо-професійною програмою

Спеціальність 126 Інформаційні системи та технології  
(код і назва)

ОПП Інформаційні управляючі системи та технології  
(код і назва)

**ЗАТВЕРДЖУЮ**

В.о.завідувача кафедри

О.А.Павлов  
(підпис) (ініціали, прізвище)

«\_\_» \_\_\_\_\_ 2019 р.

**ЗАВДАННЯ  
на магістерську дисертацію студенту  
Клименку Вадиму Михайловичу**

(прізвище, ім'я, по батькові)

1. Тема дисертації СКЛАДАННЯ ЕНЕРГЕТИЧНО ЕФЕКТИВНИХ  
РОЗКЛАДІВ В ОПЕРАТИВНО – КАЛЕНДАРНОМУ ПЛАНУВАННІ  
науковий керівник дисертації Сперкач Майя Олегівна, к.т.н., доцент  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “ 28 ” жовтня 20 19 р. № 3770-с

2. Строк подання студентом дисертації “ 2 ” 12 20 19 р.

3. Об'єкт дослідження оперативно-календарне планування підприємства

4. Перелік завдань, які потрібно розробити виконати аналітичний огляд  
існуючих систем планування, моделей планування, методів складання  
календарних планів, систем оперативно – календарного планування  
виробництва; розробити метод розв'язання задачі складання енергетично  
ефективного розкладу при виконанні робіт паралельними пристроями;

*дослідити ефективність розробленого методу побудови розкладів; розробити програмну реалізацію розробленого методу; здійснити експериментальне дослідження отриманих результатів.*

5. Орієнтовний перелік ілюстративного матеріалу *Схема структурна розгортання, схема структурна пакетів, схема структурна компонентів, схема структурна бази даних, екранні форми.*

6. Орієнтовний перелік публікацій *тези доповідей: МОДС 2019, Матеріали III всеукраїнської науково-практичної конференції молодих вчених та студентів «Інформаційні системи та технології управління» (ІСТУ-2019); стаття, журнал «Вісник Вінницького політехнічного інституту».*

#### 7. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

8. Дата видачі завдання “ 2 ” вересня 20 19 р.

#### Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	<i>Систематизація результатів огляду літе-ри</i>	<i>02.09</i>	
2	<i>Порівняльний аналіз існуючих методів розв'язання задачі</i>	<i>15.09</i>	
3	<i>Постановка та формалізація математичної моделі задачі</i>	<i>05.10</i>	
4	<i>Розробка інформаційного та програмного забезпечення</i>	<i>10.10</i>	
5	<i>Проведення експериментальних досліджень розроблених алгоритмів</i>	<i>15.10</i>	
6	<i>Розробка стартап-проекту</i>	<i>25.10</i>	
7	<i>Оформлення документації</i>	<i>12.11</i>	
8	<i>Подання роботи на попередній захист</i>	<i>20.11</i>	
9	<i>Подання роботи на основний захист</i>	<i>02.12</i>	

Студент

\_\_\_\_\_  
(підпис)

*В.М. Клименко*

\_\_\_\_\_  
(ініціали, прізвище)

Науковий керівник

\_\_\_\_\_  
(підпис)

*М.О. Сперкач*

\_\_\_\_\_  
(ініціали, прізвище)

## РЕФЕРАТ

Магістерська дисертація: 100 с., 13 рис., 22 табл., 76 джерел, 1 додаток.

**Актуальність.** Для світу питання збереження енергії є дуже важливим. Існує багато сфер промисловості, для виробничих потужностей яких, існуюча кількість енергії є недостатньою, тому необхідно оптимізовувати виробництво задля мінімізації кількості енергії, що використовується. Як відомо, задачі, в яких необхідно мінімізувати витрати енергії, привертають величезну увагу дослідників з усього світу. Дана зацікавленість пов'язана з бурхливим зростанням промисловості та їх виробничих потужностей, а також зі стрімким розвитком комп'ютерних технологій, які дозволяють прискорювати розв'язання задач за рахунок використання додаткової енергії. Але високі швидкості зростання виробництва призводять до надмірного використання енергії, що в свою чергу сприяє збільшенню витрат підприємств. Тому виникає необхідність у складанні енергетично ефективних розкладів роботи виробництва, що допоможе зменшити фінансові витрати та запобігти глобальній світовій проблемі надмірного використання невідновлюваних енергетичних ресурсів.

У зв'язку з цим, актуальним є розробка програмного продукту для складання енергетично ефективних розкладів паралельними машинами зі змінною продуктивністю, який зменшить сумарне використання енергії під час виконання робіт.

**Зв'язок роботи з науковими програмами, планами, темами.** Робота виконувалась на кафедрі автоматизованих систем обробки інформації та управління Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського» в рамках теми «Ефективні методи розв'язання задач теорії розкладів»» (№ ДР 0117U000919)

**Мета дослідження** – підвищення ефективності функціонування виробничих систем за рахунок складання оптимальних або близьких до оптимальних за енергетичним критерієм календарних планів виконання робіт з мінімізацією загальної кількості використаної енергії паралельними пристроями.

Для досягнення мети необхідно виконати наступні **завдання**:

- виконати аналітичний огляд існуючих систем планування, моделей планування, методів складання календарних планів, систем оперативно-календарного планування виробництва;
- розробити метод розв’язання задачі складання енергетично ефективного розкладу при виконанні робіт паралельними пристроями;
- дослідити ефективність розробленого методу побудови розкладів;
- розробити програмну реалізацію розробленого методу;
- здійснити експериментальне дослідження отриманих результатів.

**Об’єкт дослідження** – оперативно-календарне планування підприємства.

**Предмет дослідження** – складання енергетично ефективних розкладів в оперативно-календарному плануванні.

### **Наукова новизна одержаних результатів**

Розроблено алгоритм складання енергетично ефективних розкладів в оперативно – календарному плануванні. Створений алгоритм дозволяє зменшити кількість енергії, що використовується під час виробництва, за рахунок мінімізації щільності часових інтервалів.

**Публікації.** Матеріали роботи опубліковані в тезах 14-ї Міжнародної науково-практичної конференції «МОДС 2019» [1]; матеріалах III всеукраїнської науково-практичної конференції молодих вчених та студентів «Інформаційні системи та технології управління» (ІСТУ-2019) [2]; стаття подана до друку в журнал «Вісник Вінницького політехнічного інституту» [3].

**ЕНЕРГЕТИЧНО ЕФЕКТИВНА ЗАДАЧА, СКЛАДАННЯ РОЗКЛАДІВ, НЕВІДНОВЛЮВАНИЙ РЕСУРС, КАЛЕНДАРНИЙ ПЛАН, ПАРАЛЕЛЬНІ ПРИСТРОЇ, МІНІМІЗАЦІЯ СУМАРНОГО ВИКОРИСТАННЯ РЕСУРСУ, ДИРЕКТИВНИЙ ТЕРМІН.**

## ABSTRACT

Master's Dissertation: 100 pp., 13 fig., 22 tables, 76 sources, 1 appendixes.

### **Relevance.**

For the world, the issue of energy conservation is very important. There are many areas of industry for whose production capacities, the existing amount of energy is not enough, therefore it is necessary to optimize production to minimize the amount of energy used. As you know, tasks in which it is necessary to minimize energy costs attract huge attention of researchers from around the world. This interest is associated with the rapid growth of industry and their production capacities, as well as with the rapid development of computer technology, allowing to accelerate the solution of problems through the use of additional energy. But high production growth rates lead to excessive use of energy, which in turn contributes to increased costs for enterprises. Therefore, there is a need for energy-efficient production schedules, will help reduce financial costs and prevent the global world problem of excessive use of non-renewable energy resources.

In this regard, it is urgent to develop a software product for compiling energy-efficient schedules by parallel machines with variable productivity, which will reduce the total energy use during work.

**Relationship with working with scientific programs, plans, topics.** The work was performed at the Department of Automated Information Processing and Management Systems of the National Technical University of Ukraine «Igor SikorskyKyiv Polytechnic Institute» within the topic «Effective methods for solving the problems of scheduling theory»» (No. DP 0117U000919).

**The purpose of the** study is to increase the efficiency of production systems by drawing up optimal or close to the optimal by energy criterion of work plans with minimization of the total amount of energy used in parallel devices.

To achieve this goal, you must complete the following **tasks**:

- perform an analytical review of existing planning systems, scheduling models, scheduling methods, production and operational scheduling systems;
- develop a method for solving the problem of compiling an energy efficient schedule when performing work on parallel devices;
- investigate the effectiveness of the developed method of scheduling;
- develop software implementation of the developed method;
- carry out an experimental study of the results obtained.

**The object of study** is the operative-calendar planning of the enterprise.

**The subject of the study** is the energy efficient scheduling for current calendar planning.

### **Scientific novelty of the obtained results**

The algorithm for compiling energy efficient schedules in operational - scheduling has been developed. The created algorithm allows to reduce the amount of energy used during production by minimizing the density of time intervals.

### **Publications.**

The materials of the work were published in the abstracts of the 14th International Scientific and Practical Conference "MODS 2019" [1]. Proceedings of the Third All-Ukrainian Scientific and Practical Conference of Young Scientists and Students "Information Systems and Technologies of Management" (ISTU-2019) [2]. The article was published in the journal «Bulletin of Vinnitsa Polytechnic Institute» [3].

ENERGY EFFICIENT TASK, SCHEDULE CREATION, NON-RENEWABLE RESOURCE, CALENDAR PLAN, PARALLEL MACHINES, MINIMIZATION OF TOTAL RESOURCE USAGE, SCHEDULE DATE.

## ЗМІСТ

<b>ВСТУП .....</b>	<b>9</b>
<b>1 ОГЛЯД ДОСЛІДЖЕНЬ В ОБЛАСТІ СКЛАДАННЯ РОЗКЛАДІВ.....</b>	<b>11</b>
1.1 Зміст та основні принципи планування діяльності на підприємстві .....	11
1.2 Оперативно-календарне планування.....	15
1.2.1 Системи оперативно-календарного планування .....	17
1.2.2 Особливості оперативно-календарного планування на підприємствах різних типів виробництва .....	22
1.3 Класифікація задач теорії розкладів .....	24
1.4 Методи вирішення задач теорії розкладів.....	27
1.5 Змістовна постановка задачі .....	32
Висновок до розділу.....	32
<b>2 РОЗВ’ЯЗАННЯ ЗАДАЧІ СКЛАДЕННЯ ЕНЕРГЕТИЧНО ЕФЕКТИВНИХ РОЗКЛАДІВ В ОПЕРАТИВНО-КАЛЕНТАРНОМУ ПЛАНУВАННІ.....</b>	<b>34</b>
2.1 Математична постановка задачі .....	34
2.2 Дослідження властивостей задачі .....	35
2.2.1 Характеристика робіт .....	36
2.2.2 Характеристика директивних термінів .....	37
2.3 Алгоритм розв’язання задачі .....	37
2.4 Приклад розв’язання задачі .....	39
Висновок до розділу.....	42
<b>3 РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТАЛЬНИХ ДОСЛІДЖЕНЬ.....</b>	<b>43</b>
3.1 Кодування класів задач.....	43



3.2 Результати експериментів роботи алгоритму .....	44
Висновок до розділу.....	48
<b>4 ОПИС ПРОГРАМНОГО ТА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ .....</b>	<b>50</b>
4.1 Засоби розробки .....	50
4.2 Архітектура програмного забезпечення .....	58
4.3 Схема структурна пакетів .....	59
4.4 Схема структурна компонентів .....	59
4.5 Схема структурна бази даних .....	59
4.6 Екранні форми .....	60
Висновок до розділу.....	61
<b>5 РОЗРОБКА СТАРТАП ПРОЕКТУ .....</b>	<b>63</b>
5.1 Опис ідеї проекту .....	66
5.2 Технологічний аудит проекту .....	68
5.3 Розрахунок економічної ефективності інноваційного проекту за методикою ЮНІДО .....	83
Висновок до розділу.....	86
<b>ЗАГАЛЬНІ ВИСНОВКИ .....</b>	<b>87</b>
<b>ПЕРЕЛІК ПОСИЛАНЬ .....</b>	<b>89</b>
<b>ДОДАТКИ .....</b>	<b>96</b>
<b>ДОДАТОК А .....</b>	<b>96</b>

## ВСТУП

**Актуальність.** Сучасний стан конкуренції на ринку збути між підприємствами змушує не лише покращувати якість виробництва, та розвивати маркетингові відділи з метою більшої впізнаваності бренду, а й заощаджувати ресурси, задля зменшення собівартості продукції. Якщо на кількість витратних матеріалів сильно вплинути не вдасться, то кількість електроенергії, що використовується під час виробництва, можна зменшити за рахунок вдало складеного розкладу.

Дослідження по оптимізації витрати енергії шляхом вибору оптимальних швидкостей виконання робіт започаткувала робота Яо, Демерс і Шенкер 1995 року, представлена на щорічному симпозіумі з основ комп'ютерних наук (FOCS 1995).

Задачі, теорії розкладів, що належать до енергетично ефективних відносяться до класу NP-повних задач. Час для вирішення NP-повної задачі в найгіршому випадку буде зростати експоненційно. Тому задля вирішення задачі на практиці використовуються наближені або евристичні методи. У зв'язку з цим, актуальною буде розробка алгоритму складання енергетично ефективного розкладу для паралельних машин, що дасть більший відсоток точності результатів з мінімальним використанням обчислювальних ресурсів.

**Зв'язок роботи з науковими програмами, планами, темами.** Робота виконувалась на кафедрі автоматизованих систем обробки інформації та управління Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського» в рамках теми «Ефективні методи розв'язання задач теорії розкладів» (№ ДР 0117U000919).

**Мета дослідження** – підвищення ефективності функціонування виробничих систем за рахунок складання оптимальних або близьких до оптимальних за енергетичним критерієм календарних планів виконання робіт з мінімізацією загальної кількості використаної енергії паралельними пристроями.

Для досягнення мети необхідно виконати наступні **завдання**:

- виконати аналітичний огляд існуючих систем планування, моделей планування, методів складання календарних планів, систем оперативно-календарного планування виробництва;
- розробити метод розв’язання задачі складання енергетично ефективного розкладу при виконанні робіт паралельними пристроями;
- дослідити ефективність розробленого методу побудови розкладів;
- розробити програмну реалізацію розробленого методу;
- здійснити експериментальне дослідження отриманих результатів.

**Об’єкт дослідження** – оперативно-календарне планування підприємства.

**Предмет дослідження** – складання енергетично ефективних розкладів в оперативно-календарному плануванні.

### **Наукова новизна одержаних результатів**

Розроблено алгоритм складання енергетично ефективних розкладів в оперативно – календарному плануванні. Створений алгоритм дозволяє зменшити кількість енергії, що використовується під час виробництва, за рахунок мінімізації щільності часових інтервалів.

## **1 ОГЛЯД ДОСЛІДЖЕНЬ В ОБЛАСТІ СКЛАДАННЯ РОЗКЛАДІВ**

У наш час збереження енергії є головним питанням людства, адже більшість енергії отримується шляхом використання невідновлюваних природних ресурсів таких як нафта, вугілля та природний газ. Державні програми стимулюють населення до економії енергії та сприяють використанню відновлюваних енергетичних ресурсів. Люди, компанії та організації шукають енергоефективні товари, оскільки вартість енергії зростає так як це головний ресурс.

У виробництві для стрімкого розвитку та нарощування виробничих потужностей, підприємству необхідно не лише підприємству енергоефективне обладнання, а й використовувати планування розкладів на виробництві з урахуванням якості виробництва та кількості енергії що використовується.

Пошук золотої середини між економією енергії і збереженням якості обслуговування породив новий клас задач, в яких потрібно знайти енергетично ефективні розклад.

На сьогоднішній день існує велика кількість задач теорії розкладів, частина з них є задачами з обмеженим ресурсом. Для розв'язання даного класу задач використовуються наближені алгоритми. Проте існує потреба у розробці алгоритму, що буде гарантувати більший відсоток точності, а ніж наявні в даний час алгоритми.

### **1.1 Зміст та основні принципи планування діяльності на підприємстві**

У системі управління підприємством особливе місце належить плануванню, як його основній функції. Планування стосується цілей (того, що потрібно зробити), а також засобів (того, як це зробити). Мета планування – це кінцевий стан, якого підприємство прагне досягти в певний момент у майбутньому [4].

Планування – це процес визначення цілей, які підприємство передбачає досягти за певний період, та способів їхнього досягнення. Строк планування охоплює постановку організаційних цілей, розробку загальної стратегії досягнення таких цілей і розвиток загальної ієрархії планів інтегрування та координації діяльності. В умовах

ринку підприємства самостійно здійснюють весь комплекс планової роботи. На відміну від директивного планування в умовах ринкової економіки плани підприємств мають об'єктивно обумовлений характер, зорієнтовані на задоволення потреб споживачів в тих чи інших видах продукції. Підприємства можуть самостійно визначати шляхи розвитку виробництва, використовувати системи мотивації праці, встановлювати виробничу програму.

Поряд із широкими правами і можливостями, в ринкових умовах підприємства несуть відповідальність за результати своєї роботи, вони повинні самостійно стежити за зміною ринкової кон'юнктури, враховувати ризик. Плануючи розвиток підприємства, менеджери керуються певними правилами – тобто принципами планування. До них належать такі [4]:

- цільова спрямованість – наявність чітко визначених місії та цілей діяльності;
- системність – охоплення всіх сфер діяльності підприємства, усіх тенденцій, змін і зворотних зв'язків у ньому;
- безперервність – постійне підтримування планової перспективи, своєчасне коригування планів виходячи із одержаних сигналів, щодо зовнішніх і внутрішніх (усередині самого підприємства) змін умов господарювання;
- оптимальність використання ресурсів – максимально повна реалізація наявних резервів;
- збалансованість – відповідність між потребами в ресурсах і наявністю в таких;
- адекватність – система планування має відповідати особливостям виробничо-комерційної діяльності підприємства.

Типи планування [4]:

- техніко-економічне;
- оперативно-календарне.

Залежно від цілей діяльності підприємство будує систему планування, яка може складатися з різних видів планів.

За широтою охоплення об'єкта планування розрізняють [4]:

- стратегічні – плани, що стосуються підприємства загалом і визначають його загальні цілі, а також ті, що спрямовані на позиціонування підприємства з погляду зовнішнього середовища;
- тактичні – плани, які визначають деталі способу досягнення загальних цілей, охоплюють короткий період;
- оперативні – плани, що містять деталізовані способи досягнення цілей підприємства та його підрозділів на короткі проміжки часу (місяць, декаду, добу, зміну).

За часовим періодом розрізняють [4]:

- короткострокові (період планування менше року);
- довгострокові (строк планування, більший за рік) плани.

За частотою використання розробляють:

- одноразові плани – призначаються для задоволення потреб специфічної або унікальної ситуації;
- регулярні – плани, які забезпечують головні напрямки неодноразових дій підприємства.

За специфікою планування виділяють [4]:

- специфічні плани – які передбачають чітко окреслені цілі, наприклад, з метою збільшення обсягів виробництва на 3% маємо запровадити специфічні процедури, визначити витрати і спланувати діяльність);
- директивні – за високого ступеня невизначеності та необхідності підтримання гнучкості управління з метою реагування на неочікувані зміни ці плани визначають загальні напрямки діяльності.

План – це кількісний вираз цілей та розробка шляхів їхнього досягнення. План як це результат планування є мотивованою моделлю дій, створеною на основі кон'юнктурного прогнозу економічного середовища і поставленої мети.

План підприємства (фірми, компанії) – це розроблена система заходів, що передбачає цілі, зміст, збалансовану взаємодію ресурсів та їхній обсяг, методи і строки виконання робіт для виробництва і реалізації продукції або надання послуг. План дає змогу підприємству оцінити реальність досягнення поставлених цілей, визначити, що допомагає, а що заважає їх досягти [4].

Процес планування складається із таких етапів [4]:

- визначення цілей планування (вони є вирішальними факторами при виборі форми і методів планування);
- аналіз проблеми (визначається вихідна ситуація на момент складання плану і формується кінцева ситуація);
- пошук альтернатив (на цьому етапі серед можливих шляхів вирішення проблемної ситуації обирається найкращий та розробляються необхідні дії);
- прогнозування (формується уявлення про розвиток ситуації, яка планується);
- оцінка (проводяться оптимальні розрахунки для вибору найкращої альтернативи);
- прийняття планового рішення (обирається і оформляється єдине планове рішення).

Підприємство розглядають як багатоцільову і відкриту систему. Це є наслідком різноманітності конкретних функцій управління. Тому будь-яке підприємство має ієрархічну систему цілей. Цілі багато в чому зумовлюють характер і особливості діяльності підприємства нині і в майбутньому. Система цілей характеризується якісними та кількісними показниками, які визначають відповідні види діяльності підприємства [4].

## 1.2 Оперативно-календарне планування

Планування діяльності підприємства розподіляється на техніко-економічне й оперативно-календарне. Техніко-економічне планування охоплює розроблення планів виробництва, розвитку техніки й економіки підприємства. У процесі техніко-економічного планування встановлюють обсяги виробництва продукції, виконують розрахунки необхідних ресурсів, розробляють заходи технічного й організаційного розвитку підприємства [5].

Розвитком і продовженням техніко-економічного планування є оперативно-календарне планування. Завдання цього етапу планування – забезпечення злагодженої роботи всіх підрозділів підприємства, рівномірного виконання плану виробництва продукції у встановленому обсязі й номенклатурі з найкращим використанням виробничих ресурсів.

Оперативно-календарне планування має виконавчий характер. Його роль полягає в подальшій конкретизації плану виробництва у вигляді завдань у натуральній кількості на коротші відрізки часу й за конкретними виконавцями: цехами, ділянками та робочими місцями [5].

Оперативно – календарне планування мусить забезпечити реалізацію річних планів підприємства. Для цього виробнича програма розподіляється за часом та обсягами виконання робіт із подальшим доведенням сформованих завдань до кожного виробничого підрозділу.

При оперативно – календарному плануванні необхідно дотримуватися наступних вимог [5]:

- принципу гнучкості;
- можливості оперативного корегування планових завдань внаслідок зміни загальних планових показників та умов діяльності підприємства;
- збалансованості змін для всіх підрозділів підприємства.



Підрозділи, діяльність яких не виходить за межі підприємства, розробляють лише плани виробництва та витрат. Решта показників планується на рівні підприємства.

Підрозділи, що самостійно діють на ринку мають більш розвинену структуру планів, розробляють власну маркетингову стратегію і можуть тій чи іншій мірі впливати на розподіл фінансових ресурсів. Вони розробляють систему планів, що близька до планів самостійних підприємств [5].

Основними завданнями планування діяльності внутрішніх підрозділів на підприємстві є:

- забезпечення ритмічності та своєчасності виробництва запланованих обсягів і номенклатури продукції та її постачання споживачам;
- забезпечення рівномірності завантаження обладнання, працівників і виробничих площ;
- забезпечення безперервності виробництва, мінімальної тривалості виробничого циклу, що сприятиме зменшенню незавершеного виробництва та прискоренню оборотності оборотних коштів.

Основні вхідні дані, що необхідні для здійснення оперативно-календарного планування [5]:

- виробнича програма підприємства на плановий рік;
- технологія та послідовність виробничого процесу;
- норми часу для окремих операцій;
- технологічний маршрут руху заготовок, деталей, продукції;
- режими роботи цехів, виробничих дільниць;
- план регламентованих робіт з ремонту та обслуговування обладнання.

Оперативне планування на рівні підприємства здійснюється для окремих цехів, на рівні окремих цехів - для дільниць та робочих місць [5].

### 1.2.1 Системи оперативно-календарного планування

Оперативно-календарне планування передбачає календарне пов'язання у часі виробничих процесів структурних підрозділів підприємства, враховує послідовність виробництва продукції та її основних частин, тривалість циклів обробки. Воно розподіляється на календарне планування та диспетчерське регулювання [6].

Календарне планування передбачає розподіл річної виробничої програми підприємства за окремими виробничими підрозділами (цехами) та термінами виконання стосовно кожного виду продукції, а також доведення встановлених завдань до конкретних виконавців. За його допомогою розробляються добові та змінні завдання, узгоджується послідовність виконання робіт окремими виконавцями.

Календарне планування передбачає:

- розробку календарно-планових нормативів;
- складання планів-графіків руху предметів праці в часі та просторі під час виробничого процесу;
- розрахунок завантаження обладнання та виробничих площ;
- доведення виробничих завдань, що формуються на основі планів-графіків, до підрозділів, виробничих ділянок і робочих місць.

Диспетчерське регулювання забезпечує оперативне регулювання процесу виробництва через: систематичний облік та контроль за виконанням зміннодобових завдань, поточну підготовку виробництва, оперативне усунення можливих недоліків і відхилень, облік випуску продукції та використання ресурсів. Для проведення такої роботи на підприємстві може бути створений спеціальний диспетчерський відділ (служба) як на рівні підприємства так і окремих цехів (підрозділів) [6].

Основними характерними властивостями диспетчерського регулювання є:

- централізація, тобто регулювання здійснюється з єдиного центру (диспетчерського відділу), розпорядження якого обов'язкові для виконання на будь-якому рівні підприємства (від начальника підрозділу до робітника);
- оперативність, а саме прийняття ефективних рішень та розробка заходів для ліквідації можливих відхилень від запланованого процесу виробництва та наслідків впливу таких відхилень. Це здійснюється на основі контролю за ходом виконання планових завдань.

Центральний диспетчерський відділ (на рівні підприємства) виконує наступні функції [6]:

- перевіряє виконання плану випуску продукції за номенклатурою та асортиментом;
- контролює процес поповнення запасів міжцехових складів заготовками, деталями для забезпечення безперебійності виробничого процесу;
- наглядає за підготовкою до виробництва та ходом ремонту обладнання.

Диспетчерська служба на рівні цеху (підрозділу) здійснює:

- контроль за виконанням змінних та добових завдань;
- облік та аналіз незапланованих простоїв обладнання;
- контроль за ходом ремонту обладнання та профілактичними роботами.

Об'єкти диспетчерського контролю можуть бути різні в залежності від типу виробництва. Для серійного виробництва такими об'єктами є:

- терміни запуску-випуску партії одиниць продукції (деталей);
- стан запасів матеріалів та готової продукції на складах;
- ступінь комплектної забезпеченості робіт, які пов'язані зі зборкою.

Для одиничного виробництва об'єктами диспетчерського контролю є:

- своєчасна підготовка виробництва;
- терміни випуску продукції;
- стан матеріально-технічного забезпечення.

Для масового виробництва об'єктами контролю є:

- дотримання ритму поточкових ліній;
- виконання профілактичного обслуговування обладнання;
- стан та наявність лінійних заготовок.

Оперативно-календарне планування розрізняють в залежності від рівня планування: міжцехове та внутрішньоцехове [6].

Міжцехове планування передбачає встановлення цехам взаємопов'язаних виробничих завдань згідно виробничої програми підприємства та забезпечення узгодженості в їх роботі. На його основі підтримується ритмічна робота основних виробничих підрозділів, забезпечується безперебійність їх обслуговування допоміжними цехами і службами (своєчасність поставки матеріалів та запчастин, оперативність ремонтних робіт і т.п.). Для цехів планове завдання складається на квартал із розподілом за місяцями [6].

Внутрішньоцехове планування – це розподіл цехового планового завдання між дільницями, доведення його завдань до кожної виробничої дільниці та робочого місця. В ході планування розробляються також поточні графіки роботи дільниць з урахуванням річного плану випуску та річного плану реалізації продукції основних цехів. Виробничі завдання цехам у натуральному вимірі розробляються у порядку, зворотному до послідовності технологічного процесу, тобто від випускних цехів до заготівельних. У плановому завданні кожного цеху враховуються [6]:

- поставки продукції (заготовок, деталей, комплектів тощо) цехам-споживачам;
- поставки на склад готової продукції;
- можлива зміна незавершеного виробництва з метою підтримання його величини на нормативному рівні.

Структура і показники виробничих завдань залежать від діючої на підприємстві системи оперативно-календарного планування [6].

**Система оперативно-календарного планування** – це сукупність методів та способів розрахунку основних планових показників для регулювання процесу виробництва, ефективного використання робочого часу та інших ресурсів підприємства. Основні характеристики таких систем планування:

- планово-облікова одиниця;
- тривалість планового періоду;
- способи і прийоми розрахунків планових показників.

Планово-облікова одиниця – це вид продукції або комплекс робіт підрозділу, що розглядаються як кінцевий результат його роботи при плануванні, обліку, аналізі та оперативному регулюванні виробництва [7].

Планово-обліковий період – це період часу (змінa, доба, місяць, декада тощо), для якого розробляються планові завдання при оперативно-календарному плануванні.

Вид планово-облікових одиниць різний для різних цехів і типів виробництва:

- для випускних цехів – це найменування й обсяг готових виробів згідно з планом виробництва підприємства;
- для заготівельних та обробних цехів – це комплекти деталей на замовлення, виріб, складальну одиницю (при одиничному та серійному виробництвах), а також окремі найменування заготовок та деталей (при масовому виробництві).

Розглянемо найбільш розповсюджені системи оперативно-календарного планування, серед яких: позамовна, по комплектна, подетальна, за тактом випуску виробів, на склад або ринок [7].

Позамовна система. Основна планово-облікова одиниця при цій системі - це окреме виробниче замовлення, яке містить декілька однотипних робіт конкретного споживача-замовника. Термін виконання замовлення визначається на основі розрахунків тривалості виробничих циклів та нормативів випереджень, що встановлюються замовником або ринком. Основа позамовної системи планування –

цикловий наскрізний графік, де вказується черговість робіт над виробом, терміни їх виконання та виконавці. Найчастіше ця система застосовується в одиничному та дрібносерійному виробництві [7].

Покомплектна система. Планово-облікова одиниця в цій системі – це деталі, що входять до збірного вузла або загального комплекту товарів, згрупованих за певними ознаками. Календарні завдання розробляються не за деталями окремого найменування, а за групами, комплектами деталей на вузол, машину, замовлення або обсяг робіт (послуг). Завдання цехам видаються у вигляді переліку виробів (можливо за номерами), де вказується час їх комплектування деталями та складальними одиницями даного цеху. Ця система характерна для серійного та масового виробництва.

Подетальна система. Дана система найбільш ефективна в умовах масового і крупносерійного виробництва. Планово-обліковою одиницею тут виступає кожна деталь, складальна одиниця, кожне робоче місце. Календарно-оперативні плани містять показники обсягу випуску та маршрут руху деталей кожного найменування за всіма виробничими стадіями і технологічними операціями на певний плановий період [7].

Планування за тактом випуску виробів. При плануванні за цією системою обирається певна розрахункова одиниця часу (організаційний такт), за яку повинні бути виконані взаємопов'язані роботи виробничого процесу. Ця система передбачає вирівнювання у відповідності з одиничним розрахунковим часовим інтервалом тривалості технологічних операцій взаємопов'язаних робіт на всіх стадіях загального виробничого процесу. За допомогою такту визначається кількісні та календарні показники щодо періоду випуску продукції.

Планування на склад або ринок. При цій системі визначається необхідна кількість готових виробів, яка повинні постійно знаходитись на проміжних, кінцевих етапах виробництва та на складі для забезпечення безперервності виробництва і збуту

продукції. При зниженні необхідного запасу продукції до певного рівня (“точки замовлення”), він мусить бути поповнений до запланованої величини. Така система, крім “точки замовлень” передбачає також розрахунок мінімального і максимального запасів, і тому має назву системи "мінімум-максимум" [7].

### 1.2.2 Особливості оперативно-календарного планування на підприємствах різних типів виробництва

Одиничне виробництво. При даному типі виробництва об'єктом планування є замовлення. Використовуються наступні системи оперативно-календарного планування [8]:

- позамовна – при короткому циклі складання виробу (до одного місяця);
- комплектно-вузлова та комплектно-групова системи при тривалості складального циклу більше одного місяця.

Календарно-планові розрахунки містять:

- розрахунок тривалості виробничого циклу виготовлення виробу;
- визначення календарних випереджень у роботі окремих виробничих підрозділів;
- складання плану – графіка виконання замовлення;
- складання зведеного графіка виконання замовлень згідно виробничій програмі на плановий період;
- розрахунки завантаження обладнання і виробничих площ за календарними періодами (об'ємно – календарні розрахунки), корегування зведеного графіка з метою вирівнювання завантаження за окремими плановими періодами.

Розрахунки виконуються на основі тривалості виробничого циклу виготовлення виробу. При побудові циклового графіка необхідно враховувати міжцехові перерви, які мають страховий характер і становлять 3-5 днів. Виробнича програма цеху на

місяць є основою для розробки завдань для кожної ділянки та робочого місця. Вихідними даними при цьому є:

- конструкторсько – технологічні специфікації;
- карти технологічних процесів;
- карти розшифровки комплектів;
- терміни початку і закінчення робіт.

Після складання планів для ділянок проводяться календарні розрахунки стосовно обладнання та виробничих площ, встановлюється змінність роботи ділянок та окремих груп обладнання, уточнюється потреба в робітниках.

При складанні зведеного графіка здійснюються розрахунки завантаження обладнання і площ всіма видами робіт з виготовлення продукції в усіх підрозділах підприємства [8].

Серійне виробництво. Основним завданням календарного планування в даному випадку є забезпечення періодичності виробництва відповідно до плану при повному та рівномірному завантаженні обладнання, площ, робітників.

При оперативному плануванні серійного виробництва продукції важливе значення має вибір величини партії деталей. Якщо деталі виготовляються для малих серій виробів, то величина партії дорівнює їх потребі. Якщо деталі потрібні періодично або постійно, але їх обробкою обладнання повністю не завантажено, то розмір партії визначається з урахуванням конкретних виробничих умов [8].

Масове виробництво. Об'єктами планування при масовому виробництві виступають окремі деталі, складальні одиниці, окремі вироби. Для планування потокового виробництва та розробки календарно-планових нормативів розраховується кількість робочих місць на лінії, яке забезпечує виконання змінного і годинного завдання щодо кожної операції. При даному типі виробництва використовуються наступні системи оперативно – календарного планування: за тактом випуску виробів; планування на склад.



Оперативно-календарне планування на підприємствах різних типів виробництва повинно забезпечити рівномірність виробничого процесу та випуск якісної продукції у необхідній кількості та у встановлений термін [8].

Проте, незалежно від типу виробництва для складання ефективних планів використовують теорію розкладів.

### **1.3 Класифікація задач теорії розкладів**

Якщо ми зіштовхнулись з проблемою вибору черговості виконання завдань з використанням ресурсів, то ця задача належить до теорії розкладів. Проте задачі теорії розкладів не є однотипними і в свою чергу поділяються на багато класів задач [9].

Задачі теорії розкладів можна поділити на дві групи:

- задачі з перериваннями. У будь-який момент часу обслуговування вимоги на машині може бути перервано (з можливістю завершення пізніше на тій же або іншій машині) заради обслуговування іншого вимоги;
- задачі без переривань. Кожну вимогу на машині обслуговується від початку до кінця без переривань.

Існують різні варіанти задач теорії розкладів, частина з них є NP-повними, частина належить до класу поліноміальних задач (безліч задач, для яких існують «швидкі» алгоритми рішення), для частини задач так і не вдалося довести приналежність до якого-небудь класу складності. Існує гіпотеза, що задача, що допускає переривання, не складніше задачі без переривань. Для більшості задач вона дотримується, крім однієї, де для варіанту без переривання доведена її приналежність до класу поліноміальних задач, в той час як для аналогічної задачі з перериваннями не існує доказів приналежності до якого-небудь класу складності [9].

За типом цільової функції [9]:

- задачі з сумарними критеріями оптимізації. Задачі, в яких необхідно мінімізувати сумарне значення моментів закінчення обслуговування робіт;
- задачі з мінімаксними критеріями оптимізації. Відмінність цих задач від задач з сумарними критеріями полягає в тому, що потрібно мінімізувати не суму деяких значень, а лише максимальне з них;
- багатокритеріальні задачі оптимізації. Якщо в досліджуваних задачах необхідно побудувати оптимальне рішення з точки зору декількох цільових установок (функцій), то такі задачі називаються багатокритеріальний. Наприклад, якщо в згаданій задачі необхідно не тільки мінімізувати значення але мінімізувати і час простою процесора (приладу), то це багатокритеріальна задача;
- задача на побудову допустимого розкладу. Даний клас задач можна звести до оптимізаційних задач, ввівши спеціальну функцію штрафу, який потрібно мінімізувати.

За способом задання вхідної інформації [9]:

- детерміновані задачі (off-line). Для таких задач характерно, що всі вхідні дані завдання точно відомі, тобто ми маємо значення всіх параметрів до початку її рішення;
- динамічні задачі (on-line). Для даних задач розклади будуються в режимі реального часу, тобто перед початком вирішення задачі ми не знаємо значень всіх параметрів. Розклад будується частинами в міру надходження нової інформації. При цьому в будь-який момент може бути знадобитися відповідь про якість складеного «часткового» розкладу.

За типом виконання робіт на машинах [9]:

- Open shop, відкрита лінія – для кожної вимоги задана своя підмножина машин, на кожній з яких вона повинна обслуговуватися деякий час. Порядок

обслуговування на цих машинах довільний. Задаються різноманітні цільові функції;

- Job shop, робочий цех – для кожного вимоги задана своя впорядкована підмножина машин (маршрут), на яких вона повинна обслуговуватися в заданому порядку. Задаються різноманітні цільові функції;
- Flow shop, потокова лінія – всі машини впорядковані і кожна вимога проходить всі машини в цьому порядку. Розклад задано перестановкою вимог. Як правило, мінімізується загальний час обслуговування вимог;
- завдання з директивними строками. Для кожної вимоги заданий момент надходження, час обслуговування і директивний термін закінчення обслуговування. Порядок обслуговування на приладах довільний. Необхідно знайти розклад, що задовольняє директивні терміни. При існуванні такого розкладу можна ставити задачу мінімізації числа переривань.

За типом шуканого рішення [9]:

- задача упорядкування. У цих задачах вже задано розподіл робіт за виконавцями, а також визначені всі параметри робіт (тривалість виконання, час надходження і т.д.). Необхідно скласти розклад (або порядок) виконання робіт кожним виконавцем;
- задача узгодження. Основна увага в цих задачах приділяється вибору тривалості виконання робіт, часу надходження і іншим параметрам;
- задача розподілу - пошук оптимального розподілу робіт за виконавцями.

Підкласом задач упорядкування є задачі з обмеженим ресурсом. В задачах даного класу ми маємо множину машин та робіт з чітко визначеним часом надходження та директивним терміном, також кожна робота має тривалість виконання. Кожна машина може збільшувати швидкість виконання роботи, проте зі збільшенням швидкості (зменшення тривалості виконання) зростає кількість ресурсу що споживається. Необхідно мінімізувати використання ресурсу [9].

## 1.4 Методи вирішення задач теорії розкладів

Для вирішення задач теорії розкладів існує велика кількість методів дискретної оптимізації. Основні з них:

- метод динамічного програмування;
- методи теорії графів;
- метод гілок та меж;
- евристичні методи;
- ймовірнісні методи.

### Метод динамічного програмування

Динамічне програмування - розділ математичного програмування, присвячений дослідженню багатокрокових задач прийняття оптимальних рішень. При цьому багатокроковість задачі відображає реальний перебіг процесу прийняття рішень в часі, або вводиться в задачу штучно за рахунок розподілу процесу прийняття одноразового рішення на окремі етапи, кроки. Мета такого представлення полягає в зведенні вихідної задачі високої розмірності до вирішення на кожному кроці задачі меншої розмірності [10].

Нехай є деяка система  $S$ , стан якої в початковий момент часу 0 характеризується числом  $x_0$ . У кожен момент часу  $k$ ,  $k = 1, \dots, n$ , робиться певний вибір  $x_k$ , в результаті чого система змінює свій стан. При цьому кожне рішення (вибір)  $x_k$  повинно задовольняти як вихідним обмеженням системи  $S$  так і обмеженням, що виникають за рахунок раніше зроблених виборів  $x_1, x_2, \dots, x_{k-1}$ . Кожне рішення (вибір) приносить певний виграш (дохід). Для застосування схеми динамічного програмування необхідно, щоб загальний дохід за  $n$  кроків дорівнював сумі доходів від окремих кроків. Послідовність допустимих рішень (допустимого вибору) на окремих кроках зазвичай називають стратегією. Потрібно знайти серед всіх політик оптимальну, що дає максимум сумарного доходу.

В основі динамічного програмування лежить наступна проста характеристика оптимальної стратегії, сформульована Р.Беллманом в 50-і роки 20-го століття і названа принципом оптимальності Беллмана [11] : яким би не були початковий стан і перше рішення, наступні рішення складають оптимальну політику по відношенню до початкового стану, отриманого в результаті першого рішення [10].

### **Методи теорії графів**

Теорія графів – розділ математики, що вивчає властивості графів. Наочно граф можна уявити як геометричну конфігурацію, яка складається з точок (вершини) сполучених лініями (ребрами). У строгому визначенні графом називається така пара множин  $G = (V, E)$ , де  $V$  є підмножина будь-якої зліченної множини, а  $E$  – підмножина  $V \times V$  [12].

Визначення графу є настільки загальним, що цим терміном можна описувати безліч подій та об'єктів повсякденного життя. Високий рівень абстракції та узагальнення дозволяє використовувати типові алгоритми теорії графів для вирішення зовнішньо несхожих задач у транспортних і комп'ютерних мережах, будівельному проектуванні, молекулярному моделюванні тощо. Теорія графів знаходить застосування, наприклад, в геоінформаційних системах (ГІС). Існуючі або запроектовані будинки, споруди, квартали тощо розглядаються як вершини, а дороги, інженерні мережі, лінії електропередачі, що їй з'єднують, тощо – як ребра. Застосування різних обчислень, вироблених на такому графі, дозволяє, наприклад, знайти найкоротший об'їзний шлях або найближчий продуктовий магазин, спланувати оптимальний маршрут [12].

### **Метод гілок та меж**

Метод гілок та меж є загальним алгоритмічним методом вирішення різноманітних оптимізаційних задач. Він широко застосовується для таких NP-повних задач, як задача комівояжера [14] та задача про рюкзак [15]. Даний метод є варіацією

повного перебору, при якому відкидаються підмножини допустимих рішень, що не містять оптимальних значень цільової функції [13].

У методі гілок та меж використовуються дві процедури: розгалуження та знаходження оцінок (меж).

За допомогою процедури розгалуження множина допустимих рішень на кожному етапі розділяється на підмножини меншої розмірності. Ці множини стають вузлами дерева пошуку.

Процедура знаходження оцінок (меж) визначає верхні та нижні межі оптимального значення на підмножинах допустимих рішень. Якщо для знайденого вузла дерева пошуку верхня межа співпадає з нижньою, то це значення цільової функції є оптимальним і досягається на відповідній підмножині допустимих рішень [13].

### **Евристичні та ймовірнісні методи**

Незадовільний стан розвитку точних методів розв'язання задач теорії розкладів обумовив розробку наближених методів, що дозволяють отримати прийнятні розв'язки при порівняно невеликих витратах часу та засобів. Умовно наближені методи поділяються на евристичні та ймовірнісні [16].

Евристичні алгоритми засновані на прийомі, який називається прийомом зниження вимог. Він полягає у відмові від пошуку оптимального розв'язку за прийнятний час. Евристичні алгоритми використовують різні результати без строгих доведень.

Широко застосовується так званий метод локального пошуку. При цьому наперед вибрана множина перестановок використовується для послідовного покращення початкового розв'язку до тих пір, поки таке покращення можливе. У протилежному випадку виявляється досягнутим локальний оптимум.

Ще один із напрямків евристичних методів розв'язання задач теорії розкладів полягає у формуванні правил або функцій надання переваг (пріоритетів). Для кожної

$i$ -ої роботи із множини очікуючих виконання робіт, обчислюється значення функції  $f_i$  пріоритету і вибирається та робота, для якої  $f_i$  досягає максимуму чи мінімуму.

**Приклади правил пріоритетів [16]:**

- правило SPT (shortest processing time). Перевага надається тій роботі (операції) з множини готових до обробки на звільненій машині, в якій час виконання на цій машині мінімальний;
- правило LRT (longest remaining time). Вимагає вибору напруженої роботи, тобто тої, в якій сума часу виконання операцій, що залишилися, максимальна;
- правило LPT (longest processing time). Перевага надається тій роботі (операції) із множини готових до обробки на звільненій машині, в якій час виконання на цій машині максимальний.

Перевагою евристичних методів є зручність реалізації їх на ЕОМ навіть при розв'язанні громіздких задач [16].

Недоліки евристичних методів полягають у складності оцінки близькості отриманих розкладів до оптимального. Крім того, для кожної функції пріоритету існують задачі, для яких застосування даної функції приводить до поганих результатів. Один із шляхів удосконалення метода функцій пріоритетів полягає в їх прив'язці до класів задач [16].

Ймовірносні методи пов'язані з  $k$ -кратним моделюванням розкладів. Вибір робіт із множини очікуючих виконання здійснюється випадковим чином. Після  $k$ -кратного застосування вибирається найкращий розклад, який приймається за розв'язок задачі. При цьому розрізняють:

- ненаправлений випадковий пошук;
- направлений випадковий пошук без самонавчання;
- направлений випадковий пошук із самонавчанням.

Евристичні методи є дуже зручними в реалізації, адже вони не завантажують процесор обчислювальної машини, проте правильність їх на кожному кроці невідома бо не може бути підтверджена.

Перші дослідження в області енергетично ефективних задач були зроблені вченими Яо, Демерсом і Шенкером [17]. Вони представили свій метод оптимізації кількості використаної енергії шляхом вибору оптимальних швидкостей викання робіт на одній машині. Цей алгоритм отримав назву YDS та пізніше було доведено його поліноміальну складність, а також те, що його можна реалізувати за  $O(n^2 \log n)$ . Проте даний алгоритм може використовуватись лише для задач з однією машиною та робіт, що дозволяють переривання під час виконання.

У своїй роботі А.С. Анічкін та В.А. Семенов [18] надали детальний опис моделі ресурсів, які використовуються під час складання розкладів та моделі виконання робіт.

Значну частину моделей і методів управління проектами становлять задачі побудови календарних планів реалізації проекту, що пов'язані переважно з розподілом обмежених ресурсів. Задачі розподілу ресурсів на мережах належать до складних задач з багатьма екстремумами. Саме це питання розглянув Ю.О. Верес у рамках свого дослідження [19].

Також у своїй дисертації А.В. Кононов [20] описує алгоритми розв'язання енергетично ефективних задач, таких як: задача з однією машиною та перериваннями, з однією машиною – наближений алгоритм, з паралельними машинами (детерміновані та динамічні). В рамках дисертації демонструється розв'язання енергетично ефективної задачі з паралельними машинами, визначеними часом надходження робіт та директивними термінами. Проте даний алгоритм є наближеним і не дає точного результату.

І. Н. Лушакова у роботі [21] вперше розглянула задачу вибору оптимальних швидкостей приладів в двохстадійній системі з нефіксованим маршрутами за умови,



що в процесі обслуговування деяких вимог деякими приладами використовується більше половини запасу відновлюваного ресурсу.

Д.І. Архипов , А.А. Лазарєв , Г.В. Тарасов зробили дослідження в області задачі визначення завантаження ресурсів при пошуку нижніх оцінок загального часу виконання робіт проекту з урахуванням обмеження на ресурси (RCPSp). Для випадку двох ресурсів запропонований геометричний підхід до вирішення завдання і представлені два поліноміальних алгоритму точного вирішення задачі [22].

### **1.5 Змістовна постановка задачі**

Маємо множину робіт. Кожна робота має час надходження  $r_j$ , директивний термін  $d_j$  та тривалість її виконання  $W_j$  при швидкості  $S = 1$ . Необхідно розробити систему, яка дозволить складати розклади так, щоб кількість енергії витрачена під час виконання робіт була мінімальною. Для збільшення швидкості виконання роботи можна використовувати додаткову енергію, швидкість збільшується пропорційно до кількості енергії що використовують. На виході система повертає розклад з інтервалами виконання робіт та їх швидкості. Також система розраховує загальну кількість енергії витрачену під час виконання всіх робіт.

### **Висновок до розділу**

У даному розділі було зроблено огляд досліджень в області складання розкладів, розглянуто зміст та основні принципи планування діяльності на підприємстві. Було визначено основні типи планування. Також детально розглянуто оперативно – календарне планування, визначено його основні вимоги, завдання та вхідні данні. Були наведені системи оперативно – календарного планування: календарне планування, диспетчерське регулювання, міжцехове планування, внутрішньоцехове планування. Описано особливості оперативно-календарного планування на підприємствах різних типів виробництва: одиночне, серійне, масове.

Було класифіковано задачі теорії розкладів та визначено основні методи вирішення задач теорії розкладів, до них відносяться: метод динамічного програмування, методи теорії графів, метод гілок та меж, евристичні та ймовірнісні методи. Зроблено огляд досліджень за темою енергетично ефективних задач та сформовано змістовну постановку задачі.

## 2 РОЗВ'ЯЗАННЯ ЗАДАЧІ СКЛАДЕННЯ ЕНЕРГЕТИЧНО ЕФЕКТИВНИХ РОЗКЛАДІВ В ОПЕРАТИВНО-КАЛЕНТАРНОМУ ПЛАНУВАННІ

У даній роботі розглядається задача складання енергетично ефективного розкладу. У задачі відомі час надходження, директивні терміни робіт та їх тривалість виконання. Всі задачі виконуються на незалежних паралельних пристроях.

Мета: мінімізувати сумарні витрати енергії під час складання розкладу.

### 2.1 Математична постановка задачі

Задано множину робіт  $J = \{J_1, \dots, J_n\}$ , роботи цієї множини мають бути виконані на паралельних машинах  $M = \{M_1, \dots, M_m\}$ .

Для роботи  $J_j$  ( $j = 1, 2, \dots, n$ ) відомі час її надходження  $r_j$ , директивний термін  $d_j$  та тривалість  $W_j$ . Швидкість виконання робіт чи їх операцій залежить від кількості енергії, витраченої на їх виконання.

Тривалість роботи залежить від швидкості з якою вона виконується. Вибір швидкості роботи кожної машини  $S(t)$  визначається при складанні розкладу і може змінюватись протягом часу.

Нехай машина працює в інтервалі часу  $[t_0, t_1]$ . Кількість енергії, яку вона витрачає, дорівнює:

$$E_A = \int_{t_0}^{t_1} (S(T))^\alpha dt, \quad (2.1)$$

де  $\alpha > 1$  – деяка задана константа (в умовах енергетичної задачі використовуємо  $\alpha = 3$ ).

Для кожної роботи є необхідним визначити машину та інтервал, в якому вона буде виконуватися, а також швидкість її виконання так, щоб розклад був допустимим

відносно часу надходження, директивних термінів і загальна витрата енергії була мінімальною.

## 2.2 Дослідження властивостей задачі

Задачі були розбиті на підкласи задач.

Тип задачі залежить від:

- закону розподілу тривалостей;
- закону розподілу директивних строків.

Кожен розподіл має основні параметри:

- математичне сподівання (середнє значення);
- дисперсія.

Поки будемо вважати, що випадкові величини (ВВ) «тривалість роботи» та «директивний строк» мають рівномірний розподіл [1].

Відомо, що ВВ  $X$  рівномірно розподілена, якщо вона має таку функцію щільності ймовірностей:

$$f_X = \begin{cases} \frac{1}{b-a}, & \text{якщо } x \in [a; b] \\ 0, & \text{якщо } x \notin [a; b] \end{cases}$$

Умовний запис рівномірно розподіленою ВВ  $X$ :  $X \in U[a; b]$ , математичне сподівання дорівнює  $M_X = \frac{b+a}{2}$ , дисперсія –  $D_X = \frac{(b-a)^2}{12}$ .

Рівномірно розподілену ВВ  $X \in U[a; b]$  можна задавати і іншим способом, задавши середнє значення  $c = \frac{a+b}{2}$  та значення напівінтервалу  $f = \frac{b-a}{2}$  (саме так і будемо надалі задавати рівномірно розподілені ВВ).

Відмітимо, що напівінтервал співвідноситься із дисперсією так:

$$f = 3\sqrt{D \ X},$$

тому його також можна також вважати мірою розкиду ВВ [1].

### 2.2.1 Характеристика робіт

**Середнє значення тривалостей робіт** [1]. Оцінювати середню тривалість  $\bar{p}$  робіт як «S», «M» та «L» будемо з огляду на те, як вона співвідноситься із значенням директивного строку (а саме, середнім значенням директивних строків  $\bar{d}$ ).

Пропонується класифікація задач по значенню середньої тривалості робіт  $\bar{p}$ , як це показано в таблиці 2.1.

Таблиця 2.1 – Класифікація по значенню середньої тривалості

Позначення підкласу	Інтервал значень
“ $S_p$ ”	$\bar{p} \in [0.01\bar{d}; 0.05\bar{d}]$
“ $M_p$ ”	$\bar{p} \in [0.06\bar{d}; 0.1\bar{d}]$
“ $L_p$ ”	$\bar{p} \in [0.11\bar{d}; 0.15\bar{d}]$

**Примітка.** Крайні точки інтервалів підбрані з урахуванням результатів експериментів (результати, отримані для різних класів, суттєво відрізняються один від одного).

**Дисперсія тривалостей робіт** [1]. Наступним параметром, за яким проводиться класифікація задач, є дисперсія тривалості робіт, яка характеризує ступінь розкиду значень тривалостей робіт відносно середнього значення.

Значення напівінтервалу  $f_p$  для розподілу тривалостей робіт будемо задавати як відсоток від середнього значення тривалостей  $\bar{p}$ . З огляду на це, будемо розділяти задачі на три підкласи (таблиця 2.2).

Таблиця 2.2 – Класифікація задач за дисперсією тривалості

Позначення підкласу	Інтервал значень
“ $S_{fp}$ ”	$p \in [0.9\bar{p}; 1.1\bar{p}]$
“ $M_{fp}$ ”	$p \in [0.5\bar{p}; 1.5\bar{p}]$
“ $L_{fp}$ ”	$p \in [0.1\bar{p}; 1.9\bar{p}]$

### 2.2.2 Характеристика директивних термінів

Останній параметр класифікації – ступінь розсіювання директивних строків [1]. Аналогічно ступеню розсіювання тривалостей, визначимо значення напівінтервалу  $f_d$  для розподілу директивних строків робіт як відсоток від середнього значення директивних строків  $\bar{d}$  (таблиця 2.3).

Таблиця 2.3 – Класифікація задач за дисперсією директивних термінів

Позначення підкласу	Інтервал значень
“ $S_{fd}$ ”	$d \in [0.8\bar{d}; 1.2\bar{d}]$
“ $M_{fd}$ ”	$d \in [0.6\bar{d}; 1.4\bar{d}]$
“ $L_{fd}$ ”	$d \in [0.4\bar{d}; 1.6\bar{d}]$

## 2.3 Алгоритм розв’язання задачі

Було розроблено алгоритм розподілу робіт між машинами та алгоритм енергетичного розміщення робіт. За допомогою першого алгоритму ми розподіляємо роботи між машинами таким чином, щоб щільність інтервалів буда мінімальною. Після розподілу робіт, для кожної з машин виконуємо алгоритм енергетичного розміщення. Виконання даних алгоритмів дозволяє мінімізувати кількість використаної енергії за рахунок зміни швидкості виконання робіт для паралельних машин.

Для початку необхідно розподілити роботи між машинами, застосовуємо перший алгоритм:

#### **Алгоритм розподілу робіт між машинами**

**КРОК 1.** Визначимо послідовність моментів  $t_0 > t_1 > \dots > t_k$ , що відповідають часу надходження роботи та її директивному терміну.

**КРОК 2.** Знайдемо інтервали  $I(p, q) = [t_p, t_q]$  для всіх  $0 \leq p < q \leq k$ .

**КРОК 3.** Виділимо роботи, які належать до інтервалів. Робота  $J_j$  належить інтервалу  $I(p, q)$  за умови, що  $[r_j, d_j] \in I_{p,q}$ .

**КРОК 4.** Визначимо інтервал з найбільшою щільністю  $A(I_{p,q})$ .

$$A(I_{p,q}) = \frac{\sum_{J_j \in A(I_{p,q})} W_j}{|I_{p,q}|} \quad (2), \text{ де } |I_{p,q}| - \text{величина інтервалу.}$$

**КРОК 5.** Серед робіт, що належать до даного інтервалу, обираємо роботу з найбільшою тривалістю виконання  $W_j$ .

**КРОК 6.** Назначимо виконання даної роботи машині, в якій цей інтервал незаповнений, або має найменшу щільність.

**КРОК 7.** Видалимо роботу з переліку тих, що необхідно виконати. Якщо в переліку залишились ще роботи, тоді повертаємось до **КРОКУ 4**.

**КРОК 8.** Отримаємо масив усіх машин та робіт, які на них виконуються.

Після того, як всі роботи були розподілені між машинами необхідно визначити в яких інтервалах часу та з якою швидкістю вони будуть виконуватись. Для цього необхідно застосувати алгоритм енергетичного розміщення робіт на машинах:

#### **Алгоритм енергетичного розміщення робіт**

**КРОК 1.** Обираємо машину з невизначеним розкладом виконання робіт.

**КРОК 2.** Аналогічно попередньому алгоритму визначаємо інтервали та їх щільність, виконуємо **КРОКИ 1-4**.

**КРОК 3.** Визначимо роботи, які належать до даного інтервалу.

**КРОК 4.** Якщо до інтервалу належить лише одна робота  $J_j$ , виконаємо її в цьому інтервалі зі швидкістю  $W_j$ , переходимо до **КРОКУ 6**.

**КРОК 5.** Якщо до інтервалу належить більше однієї роботи, обираємо роботу з найбільшою тривалістю та перевіряємо можливі доступні для неї інтервали:

**ЯКЩО** існують інтервали з більшою величиною інтервалу, переходимо до наступної за тривалістю роботи в досліджуваному інтервалі;

**ІНАКШЕ** розміщуємо роботу в досліджуваному інтервалі.

Виконуємо дані дії, поки не пройдемо по всім роботам вказаного інтервалу.

**КРОК 6.** Визначимо швидкість виконання робіт в досліджуваному інтервалі

$$S_{p,q} = \frac{\sum_{J_j \in A(I_{p,q})} W_j}{|I_{p,q}|} \quad (2.2).$$

**КРОК 7.** Видалимо інтервал та роботи, що були визначені для виконання, із переліку на визначення. Якщо в переліку інтервалів даної машини, ще є невизначені інтервали, тоді переходимо до

**КРОКУ 2.**

**КРОК 8.** Розраховуємо кількість енергії, що необхідно для виконання робіт на машині  $M_i$ , за формулою (2.1). Якщо є ще невизначені з роботами машини, повертаємось до **КРОКУ 1**.

**КРОК 9.** Визначаємо кількість енергії затрачену при виконанні всіх робіт у складеному розкладі

$$E_{all} = \sum_m^{i=1} E_i \quad (2.3).$$

## 2.4 Приклад розв'язання задачі

Маємо 19 робіт, кожна з яких має час надходження  $r_j$ , директивний термін  $d_j$  та об'ємом  $W_j$ , та 3 машини на яких вони можуть виконуватись. У рамках виконання роботи ми можемо змінювати її швидкість за рахунок зміни кількості енергії, що



використовується на машинах. Тривалість роботи залежить від швидкості з якою вона виконується. Вибір швидкості роботи кожної машини  $S(t)$  визначається при складанні розкладу і може змінюватись протягом часу. Вхідні дані прикладу представлені в таблиці 2.4.

Таблиця 2.4 – Вхідні дані

№P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
$r_j$	3	3	4	4	4	7	5	12	7	11	12	14	15	17	25	19	23	11	13
$d_j$	12	17	16	12	11	20	12	30	17	18	19	28	22	30	30	37	31	18	34
$W_j$	7	15	18	8	15	11	16	11	31	34	27	6	21	9	37	11	16	17	15

Спочатку необхідно розподілити роботи між машинами таким чином, щоб щільність інтервалів була мінімальною. Першим кроком визначаємо всі інтервали, які є у вхідних даних.

У результаті маємо 22 інтервали: (3; 4), (4; 5), (5; 7), (7; 11), (11; 12), (12; 13), (13; 14), (14; 15), (15; 16), (16; 17), (17; 18), (18; 19), (19; 20), (20; 22), (22; 23), (23; 25), (25; 28), (28; 30), (30; 31), (31; 34), (34; 37).

Після визначення інтервалів розраховуємо щільність інтервалів  $A(I_{p,q})$  за формулою 2.2.

Визначаємо інтервал з найбільшою щільністю та роботу, що належить даному інтервалу, з найбільшим об'ємом. Після цього закріплюємо виконання роботи за машиною, в якій даний інтервал є не заповнений, або має найменшу щільність. Видаляємо розміщену роботу з загального переліку робіт, та повторюємо вищевказані дії, до тих пір поки всі роботи не будуть закріплені за машинами. Розподіл робіт між машинами представлено в таблиці 2.5.

Таблиця 2.5 – Результати розподілу робіт між машинами

№M	1	2	2	3	1	1	3	3	2	1	1	2	3	1	1	2	3	3	2
№P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
$r_j$	3	3	4	4	4	7	5	12	7	11	12	14	15	17	25	19	23	11	13
$d_j$	12	17	16	12	11	20	12	30	17	18	19	28	22	30	30	37	31	18	34
$W_j$	7	15	18	8	15	11	16	11	31	34	27	6	21	9	37	11	16	17	15

Після того як всі роботи були розподілені між машинами необхідно визначити в яких інтервалах часу та з якою швидкістю вони будуть виконуватись. Для цього необхідно застосувати алгоритм енергетичного розміщення робіт на машинах. На кожній з машин, аналогічно попередньому алгоритму визначаємо інтервал з найбільшою щільністю. Якщо на даний інтервал припадає лише одна робота, тоді виконуємо її в ньому, але якщо робіт більше ніж одна, тоді обираємо роботу з найбільшим об'ємом та перевіряємо можливі інтервали виконання для неї. Після визначення допустимих інтервалів для роботи з найбільшим об'ємом, розміщуємо її в інтервалі з найменшою щільністю і переходимо до наступної після неї за розміром роботи та виконуємо аналогічні дії. Повторюємо дану послідовність операцій для всіх робіт, що були закріплені за даною машиною. В результаті розміщення робіт в інтервалі визначаємо їх швидкість виконання  $S_{p,q}$  за формулою 2.3. В результаті отримано розклад представлений на рисунку 1.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
1маш.				1[s=7]	5[s=2.14]					6[s=11]			10[s=6.8]					11[s=13.5]			14[s=1.5]				15[s=7]								
2маш.				2[s=5]			3[s=2.57]					9[s=10.33]				12[s=6]		19[s=7.5]		16[s=1]													
3маш.				4[s=8]		7[s=2.67]					18[s=17]			8[s=3.67]			13[s=3]										17[s=2]						

Рисунок 2.1 – Результат розміщення робіт на машинах з інтервалами та швидкостями виконання

Для розрахунку кількості використаної енергії, застосуємо формулу 1. Тобто для роботи під номером 5 кількість використаної енергії складає  $E_5 = 2.14^3 * 7 \approx 68.6$  од.

Визначаємо кількість використаної енергії на машинах, для першої машини кількість енергії складає 9970.76 од., для другої – 4775.4 од., для третьої – 5940.5 од.

Загальна кількість витраченої енергії визначається за формулою 4. У результаті розв’язання задачі визначили, що загальна кількість енергії використаної під час виконання робіт на трьох паралельних машинах складає 20868.9 од. Отриманий результат є оптимальним, адже кількість енергії, яка була необхідна при початковому розподілі робіт зменшилась вдвічі.

### **Висновок до розділу**

У даному розділі була наведена математична постановка задачі. Були досліджені властивості задачі та позначення, що використовуються в алгоритмі. Також було наведено характеристики робіт та директивних термінів. Було наведено алгоритм розподілу робіт між машинами та алгоритм енергетичного розміщення робіт. Послідовно використовую даний алгоритм ми можемо скласти енергетично ефективний розклад та розрахувати кількість енергії, що буде використана. Також було наведено приклад, що демонструє роботу алгоритму. Алгоритм надає рішення, що мінімізує загальну кількість використаної енергії.

## 3 РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТАЛЬНИХ ДОСЛІДЖЕНЬ

### 3.1 Кодування класів задач

Для проведення експериментальних досліджень було виділено 27 підкласів задач в залежності від середнього значення тривалості робіт, дисперсії тривалостей та дисперсії директивних термінів.

Кодування підкласів задач детально наведено в роботі [1] та виконано наступним чином: код задачі  $X/Y/Z$ , де  $X$  – середня тривалість,  $Y$  – дисперсія тривалості,  $Z$  – дисперсія директивних термінів, кодування яких наведено в таблиці 3.1.

Таблиця 3.1 – Кодування класів задач

<b>X/Y/Z</b>	<b>Середня тривалість</b>	<b>Дисперсія тривалості</b>	<b>Дисперсія директивних термінів</b>
1/1/1	«S»	«S»	«S»
1/1/2	«S»	«S»	«M»
1/1/3	«S»	«S»	«L»
1/2/1	«S»	«M»	«S»
1/2/2	«S»	«M»	«M»
1/2/3	«S»	«M»	«L»
1/3/1	«S»	«L»	«S»
1/3/2	«S»	«L»	«M»
1/3/3	«S»	«L»	«L»
2/1/1	«M»	«S»	«S»
2/1/2	«M»	«S»	«M»
2/1/3	«M»	«S»	«L»
2/2/1	«M»	«M»	«S»

Продовження таблиці 3.1

<b>X/Y/Z</b>	<b>Середня тривалість</b>	<b>Дисперсія тривалості</b>	<b>Дисперсія директивних термінів</b>
2/2/2	«М»	«М»	«М»
2/2/3	«М»	«М»	«L»
2/3/1	«М»	«L»	«S»
2/3/2	«М»	«L»	«М»
2/3/3	«М»	«L»	«L»
3/1/1	«L»	«S»	«S»
3/1/2	«L»	«S»	«М»
3/1/3	«L»	«S»	«L»
3/2/1	«L»	«М»	«S»
3/2/2	«L»	«М»	«М»
3/2/3	«L»	«М»	«L»
3/3/1	«L»	«L»	«S»
3/3/2	«L»	«L»	«М»
3/3/3	«L»	«L»	«L»

### 3.2 Результати експериментів роботи алгоритму

Для дослідження властивостей алгоритму було проведено по 100 експериментів (загальна кількість експериментів 2700) для різної кількості робіт в задачах (від 4 до 7), різної кількості машин (від 2 до 5) та для різних підкласів задач, результатами яких було відношення кількості задач, де розроблений алгоритм дав розклад з меншою кількістю використаної енергії, ніж алгоритм для паралельних машин з визначеним часом надходження та директивними термінам.

Для перевірки роботи алгоритму складання розкладу був розроблений алгоритм генерації індивідуальних задач, який генерує набори робіт, значення тривалості виконання та директивних термінів яких визначають відповідні значення дисперсій.

Окремо для різної кількості робіт, машин та класу задач отримали різне значення відношення, що дозволяє дослідити тенденцію зміни ефективності алгоритму складання енергетично ефективного розкладу.

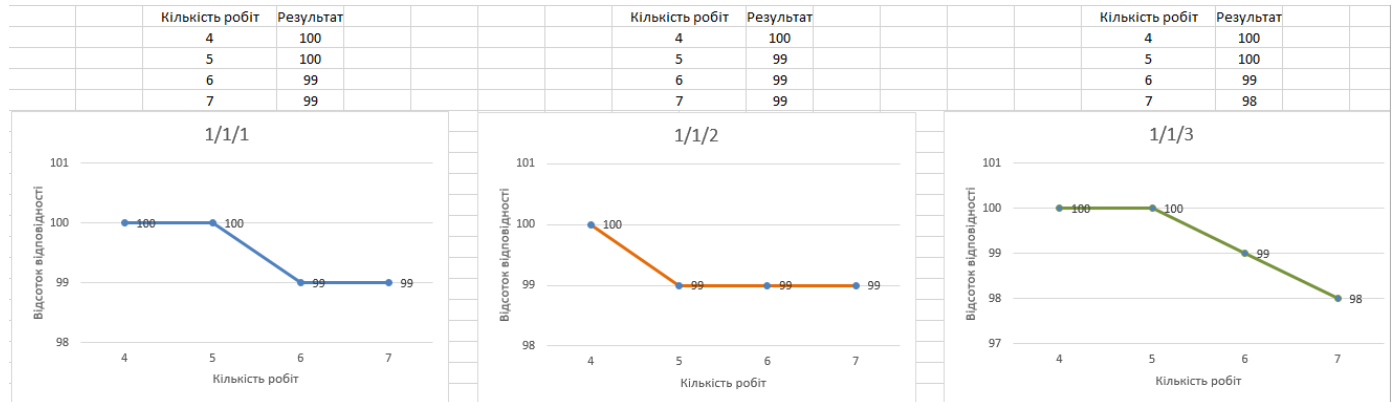


Рисунок 3.1 – Результати експериментів для задач класів 1/1/1–1/1/3

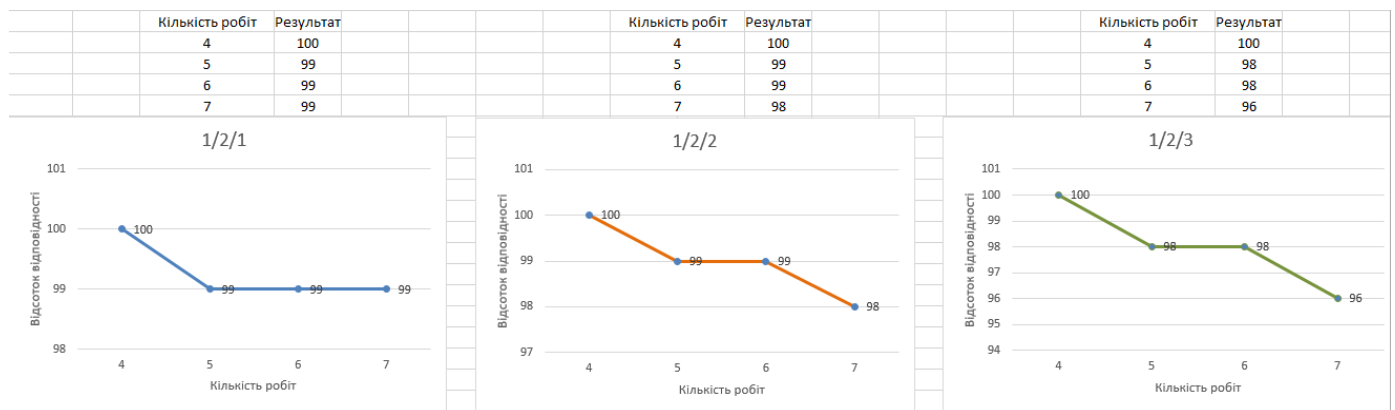


Рисунок 3.2 – Результати експериментів для задач класів 1/2/1–1/2/3

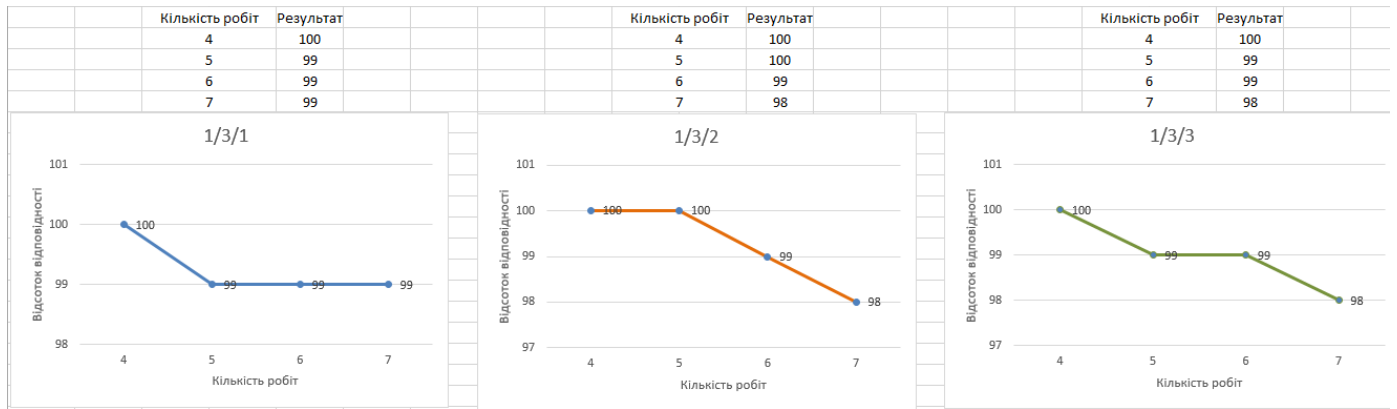


Рисунок 3.3 – Результати експериментів для задач класів 1/3/1–1/3/3

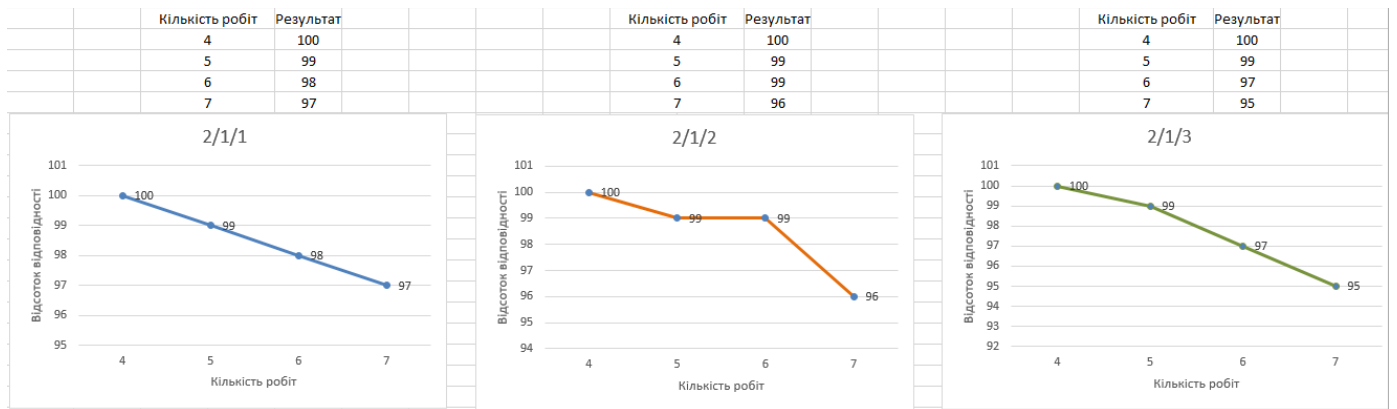


Рисунок 3.4 – Результати експериментів для задач класів 2/1/1–2/1/3

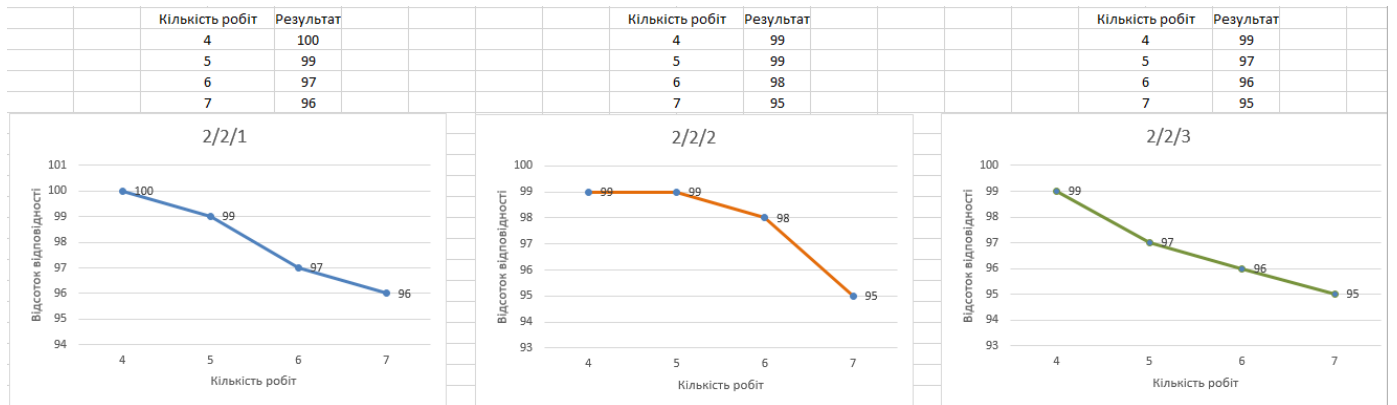


Рисунок 3.5 – Результати експериментів для задач класів 2/2/1–2/2/3

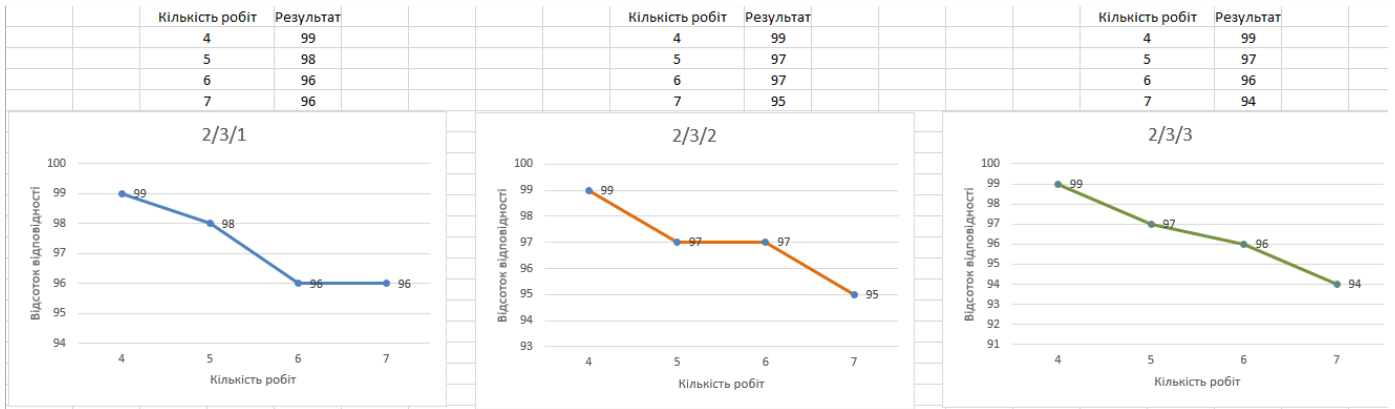


Рисунок 3.6 – Результати експериментів для задач класів 2/3/1–2/3/3

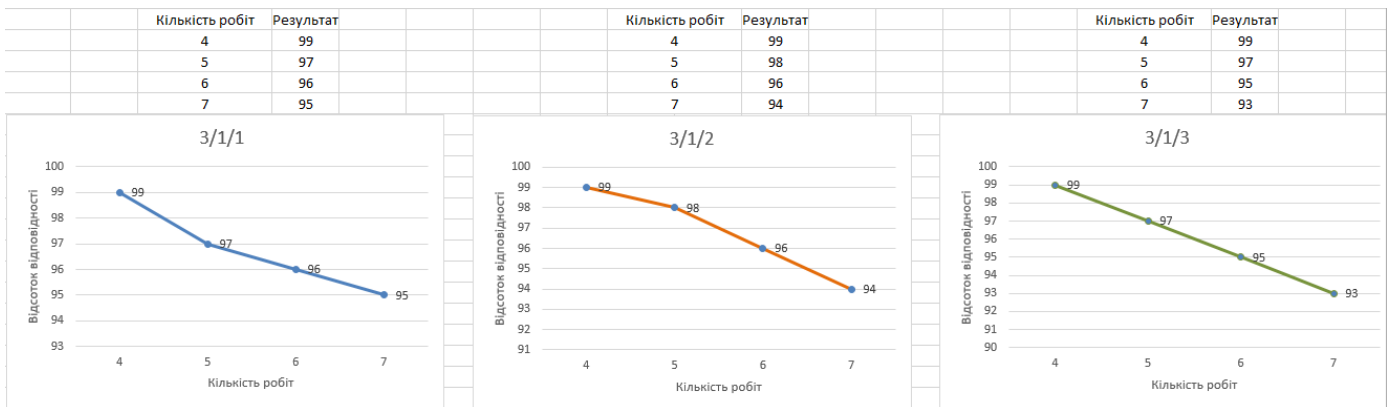


Рисунок 3.7 – Результати експериментів для задач класів 3/1/1–3/1/3

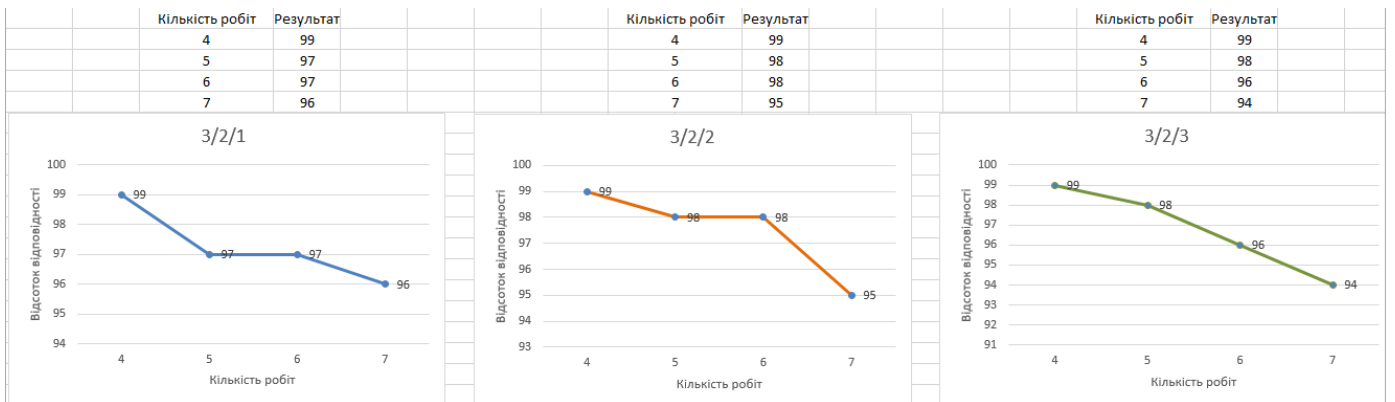


Рисунок 3.8 – Результати експериментів для задач класів 3/2/1–3/2/3



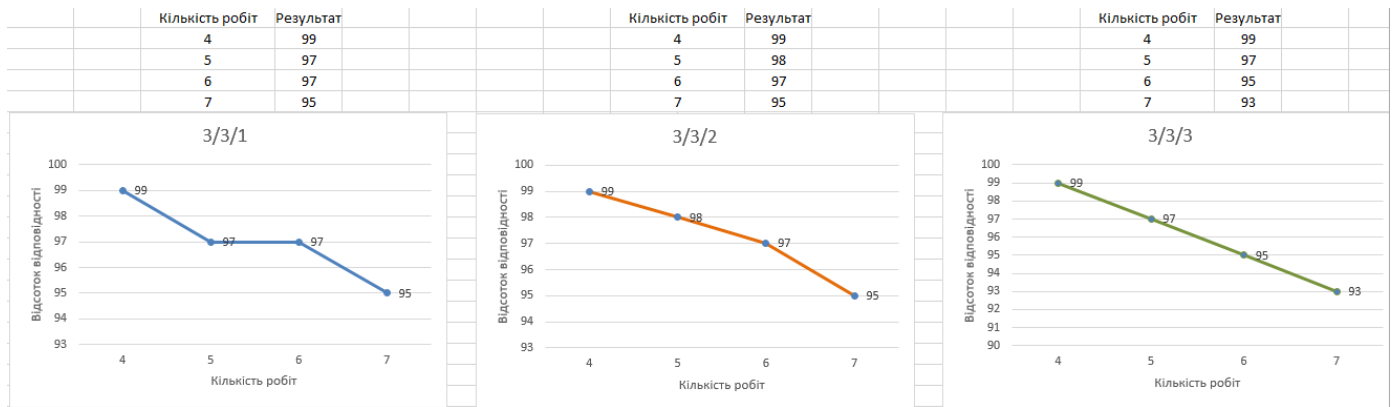


Рисунок 3.9 – Результати експериментів для задач класів 3/3/1–3/3/3

Окремо для різної кількості робіт, машин та класу задач отримали різне значення відношення, що дозволяє дослідити тенденцію зміни ефективності алгоритму складання енергетично ефективного розкладу.

При збільшенні кількості робіт якість складеного розкладу зменшується, проте при пропорційному збільшенні робіт та машин, якість розкладу залишається майже без змін. В порівнянні з алгоритмом для паралельних машин з визначеним часом надходження та директивними термінами прослідковується тенденція покращення результатів складених розкладів, тобто загальна кількість використаної енергії є меншою. Час виконання розробленого алгоритму є більшим за час виконання порівнюваного алгоритму. Також прослідковується зменшення якості складеного розкладу при збільшенні дисперсії тривалості робіт.

### Висновок до розділу

У даному розділі були проведені експериментальні дослідження. У ході проведення експериментів було визначено, що розроблений алгоритм має більшу ефективність в порівнянні з алгоритмом для паралельних машин з визначеним часом надходження та директивними термінами. Було виділено 27 класів задач з різними вхідними параметрами, середньою тривалістю робіт, дисперсією тривалості та дисперсією директивних термінів. За допомогою отриманих результатів були

сформовані графіки результатів експериментів для підкласів задач, що представлені на рисунках 3.1 – 3.9. Визначено погіршення роботи алгоритму при збільшенні дисперсії тривалості робіт.

## 4 ОПИС ПРОГРАМНОГО ТА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ

### 4.1 Засоби розробки

Під час реалізації були використані наступні програмні засоби.

**Мови програмування:** PHP, JavaScript.

**PHP** [29] – скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-сервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок (разом із Java [30], .NET [31], Perl [32], Python [33], Ruby [34]). PHP підтримується переважною більшістю хостинг-провайдерів [35]. PHP – проект відкритого програмного забезпечення.

PHP інтерпретується веб-сервером у HTML-код [36], який передається на сторону клієнта. На відміну від скриптової мови JavaScript[37], користувач не бачить PHP-коду, бо браузер отримує готовий html-код. Це є перевагою з точки зору безпеки, але погіршує інтерактивність сторінок. Але ніхто не забороняє використовувати PHP для генерування JavaScript-кодів, які виконуються вже на стороні клієнта [29].

У PHP вбудовані бібліотеки для роботи з MySQL [38], PostgreSQL [39], SQLite [40], mSQL [41], Oracle[42], dBm [43], Hyperware[44], Informix [45], InterBase [46], Sybase [47].

Завдяки стандарту відкритого інтерфейсу зв'язку з базами даних (англ. Open Database Connectivity Standard, ODBC) можна підключатися до всіх баз даних, до яких існує драйвер [29].

**JavaScript (JS)** [37] – динамічна, об'єктно-орієнтована прототипна мова програмування. Реалізація стандарту ECMAScript [48]. Найчастіше використовується для створення сценаріїв веб-сторінок, що надає можливість на стороні клієнта (пристрої кінцевого користувача) взаємодіяти з користувачем, керувати браузером, асинхронно обмінюватися даними з сервером, змінювати структуру та зовнішній вигляд веб-сторінки.

JavaScript класифікують як прототипну (підмножина об'єктно-орієнтованої), скриптову мову програмування з динамічною типізацією. Окрім прототипної, JavaScript також частково підтримує інші парадигми програмування (імперативну та частково функціональну) і деякі відповідні архітектурні властивості, зокрема: динамічна та слабка типізація, автоматичне керування пам'яттю, прототипне наслідування, функції як об'єкти першого класу.

Мова JavaScript використовується для:

- написання сценаріїв веб-сторінок для надання їм інтерактивності;
- створення односторінкових веб-застосунків (React [49], AngularJS [50], Vue.js [51]);
- програмування на стороні сервера (Node.js [52]);
- стаціонарних застосунків (Electron [53], NW.js [54]);
- мобільних застосунків (React Native [55], Cordova [56]);
- сценаріїв в прикладному ПЗ (наприклад, в програмах зі складу Adobe Creative Suite [57] чи Apache JMeter [58]);
- всередині PDF-документів тощо.

**Ajax** (Asynchronous JavaScript And XML) [59]— підхід до побудови користувацьких інтерфейсів веб-застосунків, за яких веб-сторінка, не перезавантажуючись, у фоновому режимі надсилає запити на сервер і сама звідти довантажує потрібні користувачу дані. AJAX — один з компонентів концепції DHTML [60].

Про AJAX заговорили після появи в лютому 2005-го року статті Джесі Джеймса Гарретта (Jesse James Garrett) «Новий підхід до веб-застосунків».

AJAX – це не самостійна технологія, а швидше концепція використання декількох суміжних технологій. AJAX-підхід до розробки, який призначений для користувачів інтерфейсів, комбінує кілька основних методів і прийомів:

- використання DHTML для динамічної зміни змісту сторінки;

- використання XMLHttpRequest [61] для звернення до сервера «на льоту», не перезавантажуючи всю сторінку повністю;
- альтернативний метод — динамічне підвантаження коду JavaScript в тег <SCRIPT> з використанням DOM, що здійснюється із використанням формату JSON [62]);
- динамічне створення дочірніх фреймів.

Використання цих підходів дозволяє створювати набагато зручніші веб-інтерфейси користувача на тих сторінках сайтів, де необхідна активна взаємодія з користувачем. AJAX – асинхронний, тому користувач може переглядати далі контент сайту, поки сервер все ще обробляє запит. Браузер не перезавантажує web-сторінку і дані посилаються на сервер без візуального підтвердження (крім випадків, коли ми самі захочемо показати процес з'єднання з сервером). Використання AJAX стало популярним після того, як компанія Google почала активно використовувати його при створенні своїх сайтів, таких як Gmail, Google Maps [63] і Google Suggest [64]. Створення цих сайтів підтвердило ефективність використання даного підходу.

### **Система керування базами даних: MySQL.**

**MySQL** [38] – вільна система керування реляційними базами даних.

MySQL був розроблений компанією «ТсХ» для підвищення швидкодії обробки великих баз даних. Ця система керування базами даних (СКБД) з відкритим кодом була створена як альтернатива комерційним системам. MySQL з самого початку була дуже схожою на mSQL, проте з часом вона все розширювалася і зараз MySQL – одна з найпоширеніших систем керування базами даних. Вона використовується, в першу чергу, для створення динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування.

MySQL – компактний багатопотоковий сервер баз даних. Характеризується високою швидкістю, стійкістю і простотою використання.

MySQL вважається гарним рішенням для малих і середніх застосувань. Сирцеві коди сервера компілюються на багатьох платформах. Найповніше можливості сервера виявляються в UNIX-системах [65], де є підтримка багатопоточності, що підвищує продуктивність системи в цілому.

Можливості сервера MySQL:

- простота у встановленні та використанні;
- підтримується необмежена кількість користувачів, що одночасно працюють із БД;
- кількість рядків у таблицях може досягати 50 млн;
- висока швидкість виконання команд;
- наявність простої і ефективної системи безпеки.

**Середовище розробки: PhpStorm.**

JetBrains PhpStorm [66] – комерційне крос-платформове інтегроване середовище розробки для PHP, яке розробляється компанією JetBrains на основі платформи IntelliJ IDEA [67].

PhpStorm являє собою інтелектуальний редактор для PHP [29], HTML [36] і JavaScript [37] з можливостями аналізу коду на льоту, запобігання помилок у сирцевому коді і автоматизованими засобами рефакторинга для PHP і JavaScript. Автодоповнення коду в PhpStorm підтримує специфікацію PHP 5.3/5.4/5.5/5.6/7.0/7.1 (сучасні і традиційні проекти), включаючи генератори, співпрограми, простори імен, замикання, типажі і синтаксис коротких масивів. Присутній повноцінний SQL-редактор з можливістю редагування отриманих результатів запитів.

PhpStorm розроблений на основі платформи IntelliJ IDEA, написаної на Java. Користувачі можуть розширити функціональність середовища розробки за рахунок установки плагінів, розроблених для платформи IntelliJ, або написавши власні плагіни.

Вся функціональність WebStorm[68] включена в PhpStorm:

- PhpStorm надає багатий і інтелектуальний редактор коду для PHP з підсвіткою синтаксису, розширеною конфігурацією форматування коду, перевіркою наявності помилок на льоту і розумним автодоповненням;
- підтримка PHP 5.3, 5.4 та 5.5, включаючи генератори, співпрограми, простори імен, замикання, типажі, синтаксис коротких масивів, доступ до члена класу при інстанціюванні, розіменування масиву при виклику функції, бінарні літерали, вираження в статичних виклики тощо. PhpStorm може використовуватися як для сучасних, так і для традиційних проектів на PHP;
- автодоповнення коду фіналізують класи, методи, імена змінних, ключові слова PHP, а також широко використовувані імена полів і змінних залежно від їхнього типу;
- підтримка стандартів оформлення коду (PSR1/PSR2, Drupal, Symfony2, Zend);
- підтримка PHPDoc. PhpStorm надає відповідне автодоповнення коду, засноване на анотаціях `@property`, `@method` і `@var`;
- детектор дубльованого коду;
- PHP Code Sniffer (phpcs), котрий перевіряє код на льоту;
- рефакторинги (перейменування, введення змінної/константи/поля, вбудовування змінної);
- підтримка редагування шаблонів Smarty (підсвічування синтаксичних помилок, автодоповнення функцій і атрибутів Smarty, автоматична вставка парних дужок, лапок і закриваючих тегів тощо);
- MVC подання для фреймворків Symfony2[69] і Yii [70];
- розпізнавання коду, запакованого в PHAR-архіви.

Середовище розробки:

- підтримка SQL і баз даних ( Рефакторинг схеми бази даних, генерація скриптів міграції схеми, експорт результатів виконання запиту у файл або буфер обміну, редагування збережених процедур і багато іншого;
- віддалене розгортання додатків і автоматична синхронізація з використанням FTP, SFTP, FTPS та інших протоколів;
- інтеграція з системами управління версіями ( Git - включаючи спеціальний функціонал для роботи з GitHub , Subversion , Mercurial , Perforce , CVS , TFS), що дозволяє робити багато дій, наприклад commit, merge, diff та інші, прямо з PhpStorm;
- локальна історія (Local History) (локально відстежує будь-які зміни в коді);
- PHP UML [71] (Діаграми класів UML для PHP коду з рефакторингом, що викликаються прямо з діаграми);
- підтримка Phing (надає автодоповнення, перевірку стандартних тегів, властивостей, імен цілей, значень атрибутів шляху в компоновальних файлах (build files);
- інтеграція з системами відстеження помилок;
- підтримка Vagrant, SSH консолі і віддалених інструментів;
- підтримка Google App Engine For PHP;
- PhpStorm також дозволяє різні поєднання клавіш для підвищення ефективності.

### **Представлення: HTML.**

**Hypertext Markup Language** (HTML) [36] – стандартна мова розмітки для створення веб-сторінок і веб-додатків . З Cascading Style Sheets (CSS) [72] і JavaScript, вона утворює тріаду основних технологій для World Wide Web.

Веб-браузери отримують HTML-документи з веб-сервера або з локальної пам'яті і передають документи в мультимедійні веб-сторінки. HTML описує



структуру веб-сторінки семантично і спочатку включені сигнали для зовнішнього вигляду документа.

Елементи HTML є будівельними блоками сторінок HTML. За допомогою конструкцій HTML, зображення та інші об'єкти, такі як інтерактивні форми, можуть бути вбудовані у візуалізовану сторінку. HTML надає засоби для створення структурованих документів, позначаючи структурну семантику тексту, наприклад заголовки, абзаци, списки, посилання, цитати та інші елементи. Елементи HTML окреслені тегами , написаними з використанням кутових дужок . Теги, такі як і безпосередньо вводять вміст на сторінку. Інші теги, такі як `<img />` `<input />` `<p>` оточують і надають інформацію про текст документа і можуть включати інші теги як під-елементи. Браузери не показують теги HTML, але використовують їх для інтерпретації вмісту сторінки.

HTML може вбудовувати програми, написані на мові сценаріїв, наприклад JavaScript, що впливає на поведінку та вміст веб-сторінок. Включення CSS визначає вигляд і компонування вмісту. World Wide Web Consortium (W3C), які супроводжують як HTML і CSS стандартів, заохочує використання CSS над явним презентаційним HTML з 1997 року.

HTML впроваджує засоби для:

- створення структурованого документа шляхом позначення структурного складу тексту: заголовки, абзаци, списки, таблиці, цитати та інше;
- отримання інформації із Всесвітньої мережі через гіперпосилання;
- створення інтерактивних форм;
- включення зображень, звуку, відео, та інших об'єктів до тексту.

**Представлення: CSS.**

CSS (англ. Cascading Style Sheets, укр. Каскадні таблиці стилів) [72] — спеціальна мова, що використовується для опису зовнішнього вигляду сторінок, написаних мовами розмітки даних.

Найчастіше CSS використовують для візуальної презентації сторінок, написаних HTML та XHTML, але формат CSS може застосовуватися до інших видів XML-документів.

Специфікації CSS були створені та розвиваються Консорціумом Всесвітньої мережі.

CSS має різні рівні та профілі. Наступний рівень CSS створюється на основі попередніх, додаючи нову функціональність або розширюючи вже наявні функції. Рівні позначаються як CSS1, CSS2 та CSS3. Профілі – сукупність правил CSS одного або більше рівнів, створені для окремих типів пристроїв або інтерфейсів. Наприклад, існують профілі CSS для принтерів, мобільних пристроїв тощо.

CSS (каскадна або блочна верстка) прийшла на заміну табличній верстці веб-сторінок. Головна перевага блочної верстки – розділення змісту сторінки (даних) та їхньої візуальної презентації.

CSS використовується авторами та відвідувачами веб-сторінок, щоб визначити кольори, шрифти, верстку та інші аспекти вигляду сторінки. Одна з головних переваг – можливість розділити зміст сторінки (або контент, наповнення, зазвичай HTML, XML або подібна мова розмітки) від вигляду документу (що описується в CSS).

Таке розділення може покращити сприйняття та доступність контенту, забезпечити більшу гнучкість та контроль за відображенням контенту в різних умовах, зробити контент більш структурованим та простим, прибрати повтори тощо. CSS також дозволяє адаптувати контент до різних умов відображення (на екрані монітора, мобільного пристрою (КПК), у роздрукованому вигляді, на екрані телевізора, пристроях з підтримкою шрифту Брайля або голосових браузерів та ін.).

Один і той самий HTML або XML документ може бути відображений по-різному залежно від використаного CSS. Стили для відображення сторінки можуть бути:

Стили автора (інформація надана автором сторінки):

- зовнішні таблиці стилів (англ. stylesheet), найчастіше окремий файл або файли .css;
- внутрішні таблиці стилів, включені як частина документу або блоку;
- стилі для окремого елемента.

Стили користувача:

- локальний .css-файл, вказаний користувачем для використання на сторінках і вказаний в налаштуваннях браузера (наприклад, Opera);
- стилі переглядача (браузера);
- стандартний стиль переглядача, наприклад стандартні стилі для елементів, визначені браузером, використовуються коли немає інформації про стиль елемента або вона неповна.

Стандарт CSS визначає порядок та діапазон застосування стилів, тобто, в якій послідовності і для яких елементів застосовуються стилі. Таким чином, використовується принцип каскадності, коли для елементів вказується лише та інформація про стилі, що змінилася або не визначена загальнішими стилями.

## **4.2 Архітектура програмного забезпечення**

Архітектура клієнт-сервер [73] є одним із архітектурних шаблонів програмного забезпечення та є домінуючою концепцією у створенні розподілених мережних застосунків і передбачає взаємодію та обмін даними між ними. Вона передбачає такі основні компоненти:

- набір серверів, які надають інформацію або інші послуги програмам, які звертаються до них;

- набір клієнтів, які використовують сервіси, що надаються серверами;
- мережа, яка забезпечує взаємодію між клієнтами та серверами.

Сервери є незалежними один від одного. Клієнти також функціонують паралельно і незалежно один від одного. Немає жорсткої прив'язки клієнтів до серверів. Більш ніж типовою є ситуація, коли один сервер одночасно обробляє запити від різних клієнтів; з іншого боку, клієнт може звертатися то до одного сервера, то до іншого. Клієнти мають знати про доступні сервери, але можуть не мати жодного уявлення про існування інших клієнтів. Дуже важливо ясно уявляти, хто або що розглядається як «клієнт». Можна говорити про клієнтський комп'ютер, з якого відбувається звернення до інших комп'ютерів. Можна говорити про клієнтське та серверне програмне забезпечення. Нарешті, можна говорити про людей, які бажають за допомогою відповідного програмного та апаратного забезпечення отримати доступ до тієї чи іншої інформації.

Загальноприйнятим є положення, що клієнти та сервери — це перш за все програмні модулі. Найчастіше вони знаходяться на різних комп'ютерах, але бувають ситуації, коли обидві програми — і клієнтська, і серверна, фізично розміщуються на одній машині; в такій ситуації сервер часто називається локальним.

Архітектура програмного забезпечення наведена на схемі структурній розгортання, що наведена на рисунку Додатка А.1.

#### **4.3 Схема структурна пакетів**

Схема структурна пакетів наведена на рисунку Додатка А.2.

#### **4.4 Схема структурна компонентів**

Схема структурна компонентів наведена на рисунку Додатка А.3.

#### **4.5 Схема структурна бази даних**

Схема структурна бази даних наведена на рисунку Додатка А.4.

## 4.6 Екранні форми

На рисунку 4.1 представлено список результатів вже проведених експериментів, при натисканні на номер експерименту відобразиться детальна інформація. Також фіксується час та статус проведеного експерименту.

▼ 1. Експеримент

Статус: 1

10.11.2019

^ 2. Експеримент

Статус: 1

12.11.2019

Вхідні дані

#	1	2	3	4	5	6	7
rj	3	4	7	11	12	17	25
dj	12	11	20	18	19	30	30
Wj	7	15	11	34	27	9	37

Результати

Результат

#	3	6	5	1	7	2	4
Start t	11	17	18	3	25	4	12
Finish t	12	18	19	4	30	11	17
Speed	11	9	27	7	7.4	2.14	6.8

Кількість використаної енергії

$$W = (11^3)*1 + (9^3)*1 + (27^3)*1 + (7^3)*1 + (7.4^3)*5 + (2.14^3)*7 + (6.8^3)*5 = 25753.15755102$$

▼ 4. Експеримент

Статус: 1

13.11.2019

▼ 5. Експеримент

Статус: 1

13.11.2019

▼ 6. Експеримент

Статус: 0

13.11.2019

▼ 7. Експеримент

Статус: 1

13.11.2019

▼ 8. Експеримент

Статус: 1

13.11.2019

▼ 9. Експеримент

Статус: 1

13.11.2019

Кількість машин 1

#	Час надходження ( $r_j$ )	Директивний термін ( $d_j$ )	Об'єм роботи ( $W_j$ )
1	3	11	6
2	3	10	17
3	7	20	11
4	11	18	34
5			

+

Скласти розклад

Рисунок 4.2 – Введення даних для проведення розрахунків

Після введення даних та проведення розрахунків виводиться результати та загальна кількість використаної енергії, рисунок 4.3.

### Введення даних

Генерувати дані Ввести дані

#	6	3	9	1	10	5	7	4	2	8
Start t	20	11	30	10	24	18	21	12	3	25
Finish t	21	12	31	11	25	20	24	18	10	30
Speed	9	11	10	6	10	13.5	12.33	5.66	2.43	2

Кількість використаної енергії

$$W = (9^3) \cdot 1 + (11^3) \cdot 1 + (10^3) \cdot 1 + (6^3) \cdot 1 + (10^3) \cdot 1 + (13.5^3) \cdot 2 + (12.33^3) \cdot 3 + (5.66^3) \cdot 6 + (2.42^3) \cdot 7 + (2^3) \cdot 5 = 16056.9$$

Рисунок 4.3 – Результат складеного розкладу

## Висновок до розділу

У даному розділі були наведені засоби розробки програмного забезпечення.

Серверна частина розроблена з використанням мови програмування PHP, клієнтська частина розроблена з використанням HTML, CSS, Bootstrap та JavaScript з використанням Аґах. В якості реляційної бази даних була обрана база даних MySQL. Програмний код було реалізовано в середовищі для розробки PHPStorm. Також у цьому розділі була описана архітектура програмного забезпечення та представлена схема структурна розгортання. Була наведена схема структурна пакетів, що містить 4 пакети. На схемі структурній компонентів відображений взаємозв'язок між 12 компонентами програмного забезпечення. Також наведена схема структурна бази даних. В останньому підрозділі представлені екранні форми програмного продукту.

## 5 РОЗРОБКА СТАРТАП ПРОЕКТУ

Мета: спрощення процесу планування розкладів виробництва на підприємстві та мінімізація використання ресурсів.

Результат: веб застосування для планування розкладів виробництва та розрахунку вартості використаних ресурсів.

Актуальність: На сьогоднішній день, під час постійного здорожчання цін на електроенергію, мінімізація витрат у сфері виробництва є дуже актуальним питанням. Проте реалізованих програмних продуктів чи веб застосувань майже немає. Існує багато досліджень стосовно теми планування розкладів з мінімізацією сумарного випередження. Проте навіть дослідження з даної тематики не є повністю доведені до кінця.

Очевидно виникає потреба у певному ресурсі, за допомогою якого можна було б скласти оптимальний розклад для мінімізації витрат електроенергії.

Вперше термін «стартап» почав використовуватися Forbes у серпні 1976 року і Business Week у вересні 1977 року для позначення компаній з короткою історією діяльності. Поняття закріпилось в 1990-ті роки і широко розповсюдилось. Творець методики розвитку клієнтів американський підприємець Стів Бланк визначив стартапи як тимчасові структури, що існують для пошуку бізнес-моделі [74].

Автор книги «Ощадливий стартап» та ідеолог ітеративного підходу в підприємстві Ерік Ріс зазначає, що стартапом може бути названа організація, що створює новий продукт або послугу в умовах високої невизначеності.

Підприємець, венчурний капіталіст і есеїст, засновник бізнес-акселератора Y Combinator Пол Грем вважає швидке зростання головною характеристикою стартапів (4-7 % на тиждень за ключовим показником). Його думку повторює також співзасновник PayPal, перший інвестор Facebook, автор книги Від нуля до одиниці, Пітер Тіль.



Формальними критеріями для учасників рейтингів стартапів зазвичай виступають вік компанії, кількість співробітників, прибуток та її зростання, наукомісткий характер продукту, контроль засновників над компанією й оцінка потенціалу компанії експертним журі.

Однак, Пол Грем стверджує, що наявність технологічної інновації та венчурного фінансування не має значення, а малий вік не робить компанію стартапом [74].

Деякі стартапери розглядають стартапи як культурний феномен – спільні цінності всіх членів команди і відчуття значущості вкладу кожного співробітника. Вони стверджують, що збереження цієї культури дозволяє вважати команду стартапом незалежно від розміру та контролю засновників над компанією.

Універсального підходу до опису розвитку стартапів не існує. Найчастіше згадується скорочена класифікація стадій розвитку стартапів, згідно з якою стартап проходить в своєму розвитку 5 стадій: посівну стадію (seed stage), стадію запуску (startup stage), стадію зростання (growth stage), стадію розширення (expansion stage) і стадію «виходу» (exit stage) [74].

Іноді застосовується і більш розширена класифікація стадій розвитку стартапа:

**Pre-startup стадія:**

- стадія pre-seed (pre-seed stage);
- seed стадія (seed stage);
- прототип (prototype);
- робочий прототип (working prototype);
- альфа-версія проекту або продукту (alpha);
- закрыта бета-версія проекту або продукту (private beta);
- публічна бета-версія проекту або продукту (public beta).

## Запуск проекту

Запуск проекту в експлуатацію або продукту в виробництво. Робота з першими клієнтами.

### **Post-startup стадія [74]:**

- стадія зростання (growth stage);
- стадія розширення (expansion stage);
- стадія виходу (exit stage).

Наявна практика венчурних інвестицій передбачає кілька етапів фінансування стартапів, на кожному з яких компанія залучає достатньо коштів для підтримки зростання і досягнення наступного раунду інвестицій. Оскільки інвестор отримує дохід від збільшення вартості його частки в капіталі компанії, передбачається кратне зростання компанії між інвестиційними раундами, що робить стартап привабливим для нового інвестора [74].

Більшість підходів до опису етапів фінансування, з деякими варіаціями, аналогічні представленим в есе Пола Грема “Як профінансувати стартап”:

- посівні інвестиції – перший етап залучення коштів, на якому інвесторами найчастіше виступають засновники стартапу, їхні родичі або друзі. В англійській мові закріпилося скорочення 3F, яке описує перших інвесторів більшості стартапів – friends, family and fools. Початкові кошти покривають витрати команди на проживання, розробку бізнес-плану і прототипу майбутнього продукту. У виняткових випадках посівним інвестором виступає венчурний фонд – а сума інвестицій збільшується на порядок;
- ангельські інвестиції надають приватні інвестори, зацікавлені в участі в розвитку компаній. Вхідний в капітал компанії бізнес-ангел зазвичай отримує місце в раді директорів і можливість блокувати рішення засновників, які вважатиме нерозумними. На цьому етапі стартап отримує можливість

розширити штат, закінчити роботу над першою версією продукту, залучити перших клієнтів – «ранніх послідовників» [74].

Раунд «А» – залучення коштів венчурного фонду в компанію з працездатним продуктом, клієнтами та планами розвитку. Сума інвестицій значно перевищує отримані раніше, і стартап починає будувати формальну структуру і розширюватися. За раундом «А» можуть слідувати раунди «В», «С» і наступні – вони позначаються буквами латинського алфавіту.

### 5.1 Опис ідеї проекту

Таблиця 5.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Розробити веб застосування за допомогою якого можна буде планувати розклади виробництва на підприємстві з метою мінімізації витрат на ресурси	Використання підприємством, що має пристрої(машини)	Планування розкладу доставок. Економія коштів на паливо та зберігання товарів на складах, з можливістю вибору вектора економії.
	Також змінивши предметне середовище, даний проект можна використовувати на конвеєрі виробництва, станки, що використовують електроенергію	Планування розкладу роботи конвеєра. Економія коштів на електроенергію та зберігання товарів на складі.

У даний час існує методологія за алгоритмом для паралельних машин з визначеним часом надходження та директивними термінами, даний алгоритм не використовується в програмних продуктах, проте він також розв’язує задачу.

На сьогоднішній день, якщо нам необхідно збільшити швидкість виконання роботи з урахуванням кількості використаної енергії всі розрахунки виконуються вручну, використовуючи допоміжні програмні продукти такі як Kanban. На сьогоднішній день немає програмних продуктів чи веб ресурсів, які б вирішували дану проблему. З наукових напрацювань є алгоритм під назвою YDS, який працює лише з однією машиною, та головною умовою цього алгоритму є те, що роботи можуть перериватись, даний алгоритм має поліноміальну складність, а також те, що його можна реалізувати за  $O(n^2 \log n)$ .

Таблиця 5.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту (конкурент1 – алгоритм для паралельних машин з визначеним часом надходження та директивними термінами, конкурент2 - алгоритм YDS)

№ з/п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів			W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Проектний варіант	Конкурент 1	Конкурент 2			
1	Коректність спланованого розкладу	1	0.9	0.95			+
2	Робота декількома машинами	1	1	0		+	
3	Можливість обирати вектор економії	1	0	0			+
4	Середня візуалізація для	1	1	0			+
5	Період перебування на ринку	0	1	0	+		

Таким чином, за результатами аналізу сильних, слабких та нейтральних характеристик проекту, можна зробити висновок, що до сильних сторін проекту відносяться: коректність спланованого розкладу, можливість вибору вектора економії та наявність зрозумілого користувачу дизайну. Нейтральною стороною є вміння ПЗ працювати з декількома машинами.

До слабких сторін можна віднести період перебування на ринку, так як ПЗ є новим, а бренд компанії невідомим, то відсоток довіри буде доволі низьким і на початку релізу це буде негативно впливати на продажі.

## **5.2 Технологічний аудит проекту**

Розглянемо технології, які можна буде використати при реалізації ідеї проекту. Задля визначення кінцевої технології реалізації проаналізуємо наявність технологій та їх доступність.

Для розробки програмного продукту можемо використовувати мову програмування PHP та фреймворк Laravel – з відкритим кодом PHP-фреймворк, створений Taylor Otwell і призначений для розробки веб-додатків відповідно до шаблону model–view–controller. Деякі з особливостей Laravel є модульна система упакування з виділеним менеджером залежностей Composer, різні способи для доступу до реляційних баз даних, утиліти, які допомагають в розгортанні додатків і технічного обслуговування мову програмування.

Також можлива розробка на Python та фреймворку Django – високорівневий відкритий Python-фреймворк (програмний каркас) для розробки веб-систем. Названо його було на честь джазмена Джанго Рейнхардта (відповідно до музичних смаків одного зі засновників проекту). Сайт на Django будується з однієї або декількох частин, які рекомендується робити модульними. Це одна з істотних архітектурних відмінностей цього фреймворку від деяких інших. Або можна розробити продукт на Java без використання фреймворку.

Таблиця 5.3 – Технологічна здійсненність ідеї проекту

№	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
1	Веб застосування за допомогою якого можна будет планувати розклади перевезень транспортної компанії з метою мінімізації витрат на ресурси (збереження товарів на складі та паливо)	Мова програмування PHP. Framework Laravel 5.6	Остання актуальна версія PHP 7.3. Framework Laravel версії 5.6 використовує PHP > 7.2	PHP – скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-сервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок. Також використовується для реалізації алгоритмів. Laravel один з найбільш популярних фреймворків для PHP
2		Мова програмування Python. Framework Django 2.1	Остання актуальна версія Python 3.7.2. Framework Django 2.1	Python – інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією. Django один з найбільш популярних фреймворків для Python
3		Мова програмування Java 8	Остання стабільна версія Java 8	Java – об'єктно-орієнтована кросплатформенна мова програмування. В офіційній реалізації Java-програми компілюються у байт-код, який при виконанні інтерпретується віртуальною машиною для конкретної платформи
Обрана технологія реалізації ідеї проекту: Мова програмування PHP. Framework Laravel.				

Таким чином, за результатами аналізу технологічного аудиту ідеї проекту можна реалізувати проект використовуючи мову програмування PHP та Framework Laravel, оскільки автор проекту має великий досвід роботи з даними технологіями. Аналіз ринкових можливостей запуску стартап-проекту.

Таблиця 5.4 – Попередня характеристика потенційного ринку стартап-проекту

№	Показники стану ринку (найменування)	Характеристика
1	Кількість головних продуктів, од	2 од
2	Загальний обсяг продажів, грн/ум.од	485 грн/ум.од (в місяц)
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Необхідно сертифікувати як програмне забезпечення для автоматизації управління.
6	Середня норма рентабельності в галузі (або по ринку), %	78%

Таким чином за результатами аналізу ринкових можливостей запуску стартап-проекту можна зробити висновок, що даний ринок є реальним для входження за попереднім оцінюванням, адже кількість головних продуктів конкурентів - 2, отже за умови чистої конкуренції можна зайняти 33% ринку, також ринок має зростаючу динаміку та відсутні будь-які обмеження для входу. За розрахунками загальний обсяг продажів становить 485 грн/ум.од (в місяц). Було помічено зростаючу динаміку ринку та відсутність обмежень для входу на ринок. Зі специфічних вимог є необхідність отримати сертифікат, як програмне забезпечення для автоматизації управління. Середня норма рентабельності також є дуже високою, 78%.

Таблиця 5.5 – Характеристика потенційних клієнтів стартап-проекту

№	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Планування розкладу роботи машин на підприємстві з метою мінімізації кількості енергії, що використовується	Компанії, що займаються роботою з пристроями, що можуть змінювати швидкість виконання роботи за рахунок кількості використаної енергії, та бажають зменшити витрати	Кожна група клієнтів хоче мінімізувати витрати в окремих векторах чи в загальному по компанії. Для використання даного продукту необхідно придбати місячну підписку	До продукції: зрозумілий інтерфейс, та коректність складених розкладів. До компанії: своєчасна підтримка користувачів

Отже, проаналізувавши потенційних клієнтів стартап-проекту, можна зробити висновок, що цільовою аудиторією є транспортні компанії, що займаються перевезенням та зберіганням товарів на складах.

Таблиця 5.6 – Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1	Велика конкуренція на ринку	Через те, що наше ПЗ тільки виходить на ринок, буде складно конкурувати з вже існуючими	Запустити рекламу в якій будуть прозоро описані всі переваги ПЗ в порівнянні з вже існуючими



## Продовження таблиці 5.6

№	Фактор	Зміст загрози	Можлива реакція компанії
2	Небажання клієнтів змінювати ПЗ	Так як на ринку наявні програмні продукти якими вже користуються в компанії, є ймовірність того що компанії не виявлять бажання змінювати ПЗ, через недовіру до нового продукту	Окремо працювати з кожним клієнтом, телефонувати та запрошувати на презентації ПЗ, надати список переваг в порівнянні з іншим ПЗ
3	Некоректність роботи програми (баги)	Через те що продукт новий в ньому можуть зустрічатись баги, це може відвернути клієнта від нашого ПЗ	Своєчасна підтримка клієнтів, і швидке виправлення багів в роботі ПЗ

Таким чином, велика конкуренція на ринку є основним фактором загрози, наступним фактором загрози є небажання клієнтів змінювати ПЗ. В сукупності ці два фактори перешкоджають швидкому виходу на ринок. Наступним фактором загрози є можливість некоректної роботи ПЗ, проте цей фактор є незначним.

Таблиця 5.7 – Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1	Швидке захоплення передових позицій на ринку	Так як продукт має багато переваг в порівнянні з вже існуючими аналогами, можна швидко заповнити ринок.	Продовжувати покращувати продукт та надавати кваліфіковану підтримку клієнтам(щоб їх не втратити)
2	Створення супровідного ПЗ до вже існуючого	Через прилив фінансів, можна створити ще одне ПЗ також для транспортних компаній. Це сприятиме більш швидкому виходу на ринок нового ПЗ.	Продовжувати розвивати компанію, та створювати нові продукти, для отримання більших дивідендів в майбутньому
3	Збільшення обсягів продажу ПЗ	З захопленням частки ринку, збільшаться обсяги продажу ПЗ.	Збільшувати кількість реклами та зв'язків з клієнтами для ще більшого збільшення обсягів продажу. Для цього необхідно створити відділ маркетингу.

Отже, основним фактором можливостей є швидке захоплення передових позицій на ринку за рахунок переваги у функціоналі ПЗ. Також до факторів можливостей відноситься створення супровідного ПЗ та збільшення обсягів продажу за рахунок збільшення реклами та створення маркетингового відділу.

Таблиця 5.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентноспроможною)
За типом конкуренції: чиста	Через велику кількість клієнтів відбувається чиста конкуренція	Позитивно впливає, адже вплив кожної компанії на ринок є незначним і це сприяє швидкому входженню на ринок
За рівнем конкурентної боротьби: національна	Боротьба відбувається за компанії в середині країни	Позитивно впливає, так як дає змогу розширюватися на території всієї країни
За галузевою ознакою: внутрішньогалузева	Конкуренція відбувається в галузі складання розкладів перевезень	Негативно впливає, так як ринок транспортних компаній не є великим, і вони вже використовують ПЗ конкурентів. Окремо працювати з кожним клієнтом, телефонувати та запрошувати на презентації ПЗ, надати список переваг в порівнянні з іншим ПЗ.
Конкуренція за видами товарів: товарно-родова	ПЗ є схожими (субститутами), проте мають різний функціонал.	Позитивно впливає, так як різниця в функціоналі надає переваги нашому ПЗ. Продовжувати рекламувати ПЗ та описувати його переваги.

Продовження таблиці 5.8

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентно-спроможною)
За характером конкурентних переваг: нецінова	Боротьба відбувається не на ціновому рівні, а в якості та функціоналі ПЗ.	Позитивно впливає, так як різниця в функціоналі надає переваги нашому ПЗ. Продовжувати рекламувати ПЗ та описувати його переваги.
За інтенсивністю: марочна	Конкуренція на рівні бренду.	Розвивати свій бренд(займатись брендингом) для того щоб бути впізнаваними на ринку, тоді інші ПЗ цього ж бренду будуть швидше займати передові позиції на ринку.

Таким чином, виконавши ступеневий аналіз конкуренції на ринку, можна зробити висновок, що конкуренція позитивно впливає на розвиток нових компаній на ринку, конкуренція за типом є чистою, за рівнем конкурентної боротьби - національна, адже поширюється на всю територію України, за галузевою ознакою - внутрішньогалузева (лише ринок транспортних компаній), конкуренція за видами товарів є товарно-родовою, адже всі ПЗ на ринку є субститутами, за характером конкурентних переваг - нецінова, адже конкуренція відбувається на рівні якості ПЗ, а за інтенсивністю - марочна.

**Аналіз п'яти сил Портера [75]** — методика для аналізу галузей і вироблення стратегії бізнесу, розроблена Майклом Портером у Гарвардській бізнес-школі в 1979р.

П'ять сил Портера включають в себе:

- аналіз загрози появи продуктів-замінників;
- аналіз загрози появи нових гравців;

- аналіз ринкової влади постачальників;
- аналіз ринкової влади споживачів;
- аналіз рівня конкурентної боротьби.

Методикою виділяються п'ять сил, які визначають рівень конкуренції, і, отже, привабливості ведення бізнесу в конкретній галузі. Привабливість галузі, в даному контексті, має відношення до достатньої рентабельності галузі. «Непривабливою» галуззю є така, в якій поєднання сил знижує рентабельність. Найнепривабливішою є галузь, що наближається до досконалої конкуренції [75].

Портер називає ці сили «мікросередовищем», протиставляючи йому велику кількість факторів, включених у термін «макросередовище». Макросередовище складається з тих сил, які впливають на здатність компанії до обслуговування власних клієнтів і отримання прибутку. Зміна в будь-якій із сил «мікросередовища» зазвичай вимагає, щоб компанія переглянула своє місце в галузі і на ринку. Достатня привабливість галузі не означає, що будь-яка компанія в ній буде отримувати однаковий прибуток. Компанії повинні так застосовувати свої ключові компетенції, бізнес-моделі або збутові мережі, щоб отримати прибуток більший, ніж у середньому в галузі. Хорошим прикладом є ринок пасажирських авіаперевезень. У галузі, в якій рентабельність досить низька, знаходяться компанії, які за рахунок унікальних бізнес-моделей, отримують прибуток вищий, ніж у середньому в галузі (ангели бізнесу).

“Аналіз п'яти сил Портера” включає в себе три сили “горизонтальної” конкуренції: загроза появи продуктів-замінників, загроза появи нових гравців, рівень конкурентної боротьби, та обидві сили “вертикальної” конкуренції: ринкова влада постачальників і ринкова влада споживачів.

Аналіз п'яти сил Портера є тільки частиною всіх стратегічних моделей Портера. Іншими елементами є “ланцюжок доданої вартості” і “типові стратегії” (“generic strategies”) [75].

Згідно з Портером, модель п'яти сил потрібно використовувати тільки для галузі в цілому. Модель не призначена для використання для групи галузей або якоїсь частини однієї галузі. Компанія, яка веде бізнес в одній галузі, повинна розробити мінімум один «аналіз п'яти сил Портера» для цієї галузі. Портер уточнює, що для диверсифікованих компаній, основним фундаментальним питанням корпоративної стратегії є питання вибору галузей (напрямів бізнесу), в яких компанія буде конкурувати, і для кожного напрямку бізнесу повинен проводитися власний, специфічний для галузі аналіз п'яти сил. У середньому компанії з рейтингу «Global 1000» конкурують приблизно в 52 галузях (напрямах бізнесу) [75].

Таблиця 5.9 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	ПЗ що використовується Kanban та алгоритм A1	Для входження на ринок потрібно мати ПЗ яке буде мати більший функціонал в порівнянні з тим ПЗ що вже є на ринку.	-	Якщо клієнт не задоволений ціною чи якістю ПЗ він може обрати товар-субститут	Замінники можуть бути набагато дешевшими і за допомогою цього забрати частину ринку
Висновки	Інтенсивність боротьби є високою, так як транспортних компаній не так багато на ринку.	Можливості входу на ринок є й строки виходу невеликі, так як наявна велика перевага в функціоналі.	-	Клієнти диктують умови на ринку, так як вони не будуть купляти ПЗ по завищеним цінам і неперевічених компаній	Тому ми маємо обмеження для роботи на ринку через товари-замінники

Таблиця 5.10 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Ціна	Оскільки ПЗ не можна придбати, а клієнт отримує місячну підписку, то ціна є одним з засобів ведення конкурентної боротьби. Чим більш вигідною є ціна тим вірогідніший вибір клієнта. Наша ціна в 340 грн/міс є нижчою за середню ринкову 485 грн/міс.
2	Репутація бренду	Так як в Україні багато компаній-конкурентів, репутація бренду є дуже важливим фактором. Якщо клієнт використовував одне ПЗ бренду, то з високою ймовірністю він обере ще одне ПЗ цього ж бренду.
3	Кількість функціоналу	Оскільки ПЗ конкурентів є субститутом, то кількість доступного функціоналу є важливим фактором конкуренції. Адже для багатьох клієнтів наявність додаткового функціоналу є дуже важливим фактором при виборі ПЗ.
4	Якість функціоналу	Так як ПЗ конкурентів є субститутом, то якість планування розкладів доставок товарів є одним з найважливіших факторів, тому ПЗ з найякіснішим функціоналом буде мати перевагу на ринку.
5	Частка ринку	На ринку існує група компаній, які контролюють разом понад 45% ринку. Інтенсивність суперництва між діючими конкурентами при низьких темпах зростання ринку є однією з головних сил, які діють на конкуренцію. Отже частка ринку є одним з основних факторів конкуренції.

№	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
6	Унікальність позиціонування	В нас на ринку монополістична конкуренція і фактор диференціації бренду є ключовим засобом ведення конкурентної боротьби. Тому дуже важливо, створити та підтримувати унікальне позиціонування. Це створює певний захист від конкурентних зіткнень.

Розрахунок ціни на ПЗ ринку, на розробку необхідно 7 місяців, враховуючи середню заробітню плату FullStack PHP розробника - 1250\$/міс. Вартість техніки необхідної для розробки стартапу(комп'ютер та сервери) - 4000\$, оренда приміщення - 450\$/міс, також соціальні внески 22% та інші витрати 750\$/міс.

Зробивши певні підрахунки ми маємо, що вартість ПЗ складає 23075\$. Оскільки за даними 2018 року на ринку України нараховується 1621 компанія, що займаються виробництвом з урахуванням швидкості виробництва. На даний момент часу на ринку два ПЗ-конкуренти, яким користуються компанії, тому ми можемо претендувати на 33% ринку, тобто ~ 530 компаній. Для того, щоб покрити наші витрати на розробку продукту, кожна компанія має заплатити 44\$. Ми хочемо покрити витрати на розробку за 5 місяців. Тому вартість місячної підписки на ПЗ буде складати 12\$/міс ~ 340грн/міс. Середня ринкова ціна на подібне ПЗ 485 грн/міс.

Таблиця 5.11 – Порівняльний аналіз сильних та слабких сторін «Системи планування розкладів поставок транспортною компанією»

+ - компанія що використовує алгоритм YDS

\* - компанія що використовує алгоритм для паралельних машин з визначеним часом надходження та директивними термінами

№	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з «Системою планування розкладів поставок транспортною компанією»						
			-3	-2	-1	0	1	2	3
1	Ціна	12			+			*	
2	Репутація бренду	8					*	+	
3	Кількість функціоналу	16	*	+					
4	Якість функціоналу	18		+	*				
5	Частка ринку	4							+*
6	Унікальність позиціонування	14			*	+			

Таким чином, виконавши порівняльний аналіз сильних та слабких сторін ПЗ, можна зробити висновок, що кількість та якість функціоналу, унікальність позиціонування є сильною стороною продукту і це нам надає перевагу на ринку. Ціна на ПЗ є нижчою за середню ринкову проте вона не є найнижчою. Слабкими сторонами є репутація бренду, так як він новий на ринку та частка ринку.

**Традиційний метод SWOT [76]**— аналізу дозволяє провести детальне дослідження зовнішнього й внутрішнього середовища. Результатом раціонального SWOT-аналізу, спрямованого на формування загального інформаційного потенціалу, повинні з'явитися *ефективні рішення*, що стосуються відповідної реакції



(впливу) суб'єкта (слабкої, середньої й сильної) відповідно до сигналу (слабкому, середньому або сильному) зовнішнього середовища.

Відмінна риса розглянутого підходу до проведення SWOT-аналізу на підприємстві полягає у такому:

- його побудова базується на методології *системно-цільового підходу*, де основна увага акцентується на вимірюванні параметрів зовнішнього й внутрішнього середовища в просторі, у часі й з урахуванням інформаційного потенціалу;
- проведення структуризації факторів зовнішнього й внутрішнього середовища, що є універсальними для будь-якого підприємства;
- здійснення синтезу факторів зовнішнього й внутрішнього середовища, що далі, в перспективі, відобразиться у системі рішень.

**Сильні сторони** підприємства покликані забезпечити його прискорене просування до досягнення стратегічних цілей, у той час як його «слабості» викликають гальмування. Тут також природно враховувати можливості й загрози зовнішнього середовища, без яких неможливо вірно визначити сценарії розвитку організації.

**Чинність** – це властивість організаційної системи, що за умови синтезу з можливостями зовнішнього середовища забезпечує прискорене просування системи до досягнення стратегічних цілей. Особливий інтерес при цьому представляє утворення «спіралі розвитку» – траєкторії, що формується внаслідок «накладення» (синтезу) сильних сторін організаційної системи на сприятливі можливості зовнішнього середовища (з урахуванням слабких сторін організаційної системи й погроз зовнішнього середовища).

**Слабкість** – негативна властивість організації, що визначає її гальмування в процесі руху до досягнення стратегічних цілей. Гальмування стає істотним при злитті (синтезі) основних слабкостей організаційної системи з істотними погрозами

зовнішнього середовища. При цьому також не можна ігнорувати сильні сторони підприємства й можливості зовнішнього середовища. Так формується варіант песимістичного сценарію розвитку організації. Такий сценарій може трансформуватися в «спіраль краху», якщо має місце тенденція («ефект доміно») до ослаблення можливостей і сильних сторін, з одного боку, і посиленню погроз зовнішнього середовища в сполученні з наростанням слабких сторін організації.

**Можливості** – це тенденції або події в зовнішньому середовищі, при правильній відповідній реакції на які організація домагається істотного просування до поставлених стратегічних цілей.

**Загрози** – це тенденції або події в зовнішньому середовищі, які за відсутності відповідної реакції організації спричиняють значне погіршення стану організації на шляху до виконання своїх планів.

З 1990-х років застосовується нечіткий SWOT-аналіз у процесі стратегічного планування, що полягає в розділенні чинників і явищ на категорії: *сильних і слабких сторін проекту, можливостей*, що відкриваються при його реалізації, та *загроз*, пов'язаних з його здійсненням. Традиційні математичні методи, засновані на класичній логіці, є нетерпимі до неточності та необ'єктивності істини, а також до невизначеності у економічних системах. У свою чергу невизначеність системи призводить до зростання ризиків від прийняття неефективних рішень, результатом чого можуть бути негативні економічні наслідки. З цією метою виникає потреба у методах, що ґрунтуються на нечіткій логіці [76].

Таблиця 5.12 – SWOT-аналіз стартап-проекту

Сильні сторони	Слабкі сторони
1) унікальне позиціонування. 2) значний рівень диференціації бренду "СПРПТК". 3) Кількість та якість функціоналу.	1) Репутація бренду. 2. Частка ринку.
Можливості	Загрози
1) Швидке захоплення передових позицій на ринку. 2) Створення супровідного ПЗ до вже існуючого. 3) Збільшення обсягів продажу ПЗ.	1) Велика конкуренція на ринку. 2) Небажання клієнтів змінювати ПЗ. 3) Некоректність роботи програми(баги). 4) Загроза підвищення цін, через зміну курсу валюти.

Таблиця 5.13 – Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива (орієнтований комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1.	Створення програми лояльності для постійних клієнтів (що купують підписку більше ніж 6 місяців) (розвиток ринку)	65%	2 м.
2.	Використання засобів стимулювання продажів ПЗ. (більш глибоке проникнення на ринок)	54%	3 м.
3.	Надання безкоштовного тестового періоду користування ПЗ. (диверсифікація)	76%	1,5 м.
4.	Запуск реклами та створення на сайті розділу з відгуками клієнтів.(розвиток товару)	47%	2,5 м.

Отже, після ознайомлення з альтернативами за ймовірністю отримання ресурсів та сроком реалізації найкращою альтернативою було обрано “Надання безкоштовного

тестового періоду користування ПЗ”.

### 5.3 Розрахунок економічної ефективності інноваційного проекту за методикою ЮНІДО

Таблиця 5.14 – Вихідні дані для визначення доходності проекту

№	Показники	Одиниця виміру	Значення
1	Витрати на обладнання	\$	3950
2	Термін роботи обладнання після вводу	кварталів	8
3	Гарантований обсяг підписок на квартал	\$	19080
4	Квартальний обсяг замовлень	шт.	1590
5	Валютний депозит	%	17
6	Фактор ризику	%	3.2
7	Інфляція на валютному ринку	%	9.3

Таблиця 5.15 – Поточні витрати на здійснення проекту(поквартально)

Витрати	1 кв. (I р.)	2 кв. (I р.)	3 кв. (II р.)	4 кв. (I р.)	1 кв. (II р.)	2 кв. (II р.)	3 кв. (II р.)	4 кв. (II р.)	Всього
Зарплата	3750	3750	3750	3750	3750	3750	3750	3750	30000
Соціальні витрати	825	825	825	825	825	825	825	825	6600
Амортизація	250	250	250	250	250	250	250	250	2000
Інші	5000	5000	5000	5000	5000	5000	5000	5000	40000
Всього	9825	9825	9825	9825	9825	9825	9825	9825	78600

Визначення обсягу грошових потоків ((чистий дохід -с/с) - (чистий дохід-

$c/c) \cdot 0,18 + \text{амортизація}) = (\text{чистий дохід} - c/c) \cdot 0,82 + \text{амортизація}),$

враховуючи оподаткування прибутку за ставкою 18%), дол.:

23075\$ - витрати на розробку ПЗ, враховуємо в 0-й квартал. 1-й квартал(I) =  $(19080 - 9825) \cdot 0,82 + 250 = 7840$

2-й квартал(I) =  $(19080 - 9825) \cdot 0,82 + 250 = 7840$

3-й квартал(I) =  $(19080 - 9825) \cdot 0,82 + 250 = 7840$

4-й квартал(I) =  $(19080 - 9825) \cdot 0,82 + 250 = 7840$

1-й квартал(II) =  $(19080 - 9825) \cdot 0,82 + 250 = 7840$

2-й квартал(II) =  $(19080 - 9825) \cdot 0,82 + 250 = 7840$

3-й квартал(II) =  $(19080 - 9825) \cdot 0,82 + 250 = 7840$

4-й квартал(II) =  $(19080 - 9825) \cdot 0,82 + 250 = 7840$

Визначення норми дисконтування проекту (d):  $d = 17\%$

Визначення чистого дисконтованого доходу (ЧДД) та чистої поточної вартості (ЧПВ)

$$\text{ЧДД} = \sum_{t=0}^n \frac{D_t - K_t}{(1 + d)^t}$$

де  $D_t$  - чисті доходи t-го періоду;  $K_t$  - витрати t-го періоду.

Таблиця 5.16. – Розрахунок чистого дисконтованого доходу проекту

Квартали	D	K	$\frac{1}{(1 + d)^t}$	$\frac{D}{(1 + d)^t}$	$\frac{K}{(1 + d)^t}$	ЧДД	ЧПВ
0 кв.	0	23075	1.00	0	23075	-23075,000	-23075.000
1 кв. (I р.)	7840	0	0.855	6700,855	0	6700,855	-16374.145
2 кв. (I р.)	7840	0	0.731	5727,226	0	5727,226	-10646.879
3 кв. (I р.)	7840	0	0.624	4895,065	0	4895,065	-5751.814
4 кв. (I р.)	7840	0	0.534	4183,816	0	4183,816	-1567.998

Продовження таблиці 5.16

Квартали	D	K	$\frac{1}{(1+d)^t}$	$\frac{D}{(1+d)^t}$	$\frac{K}{(1+d)^t}$	ЧДД	ЧПВ
1 кв. (II р.)	7840	0	0.456	3575,911	0	3575,911	2007,913
2 кв. (II р.)	7840	0	0.39	3056,335	0	3056,335	5064.248
3 кв. (II р.)	7840	0	0.333	2612,252	0	2612,252	7676.500
4 кв. (II р.)	7840	0	0.285	2232,694	0	2232,694	9909.194
Всього	62720	23075	X	32984,15	23075	9909,154	X

Термін окупності проекту (  $T_{ок}$  ) визначається на підставі попередніх розрахунків ЧДД та ЧПВ (табл. 16):

де  $p$  - останній рік, коли  $ЧПВ < 0$ ;

$ЧПВ_p$  - значення ЧПВ в  $p$ -му році (без мінусу);  $ЧДД_{p+1}$  - значення ЧДД в  $(p+1)$ -му році.

$T_{ок} = 4 + 1567.998/3575,911 = 4,4385$  кварталу = 0,4385 кварталу (II року). Термін окупності проекту дорівнює 0,4385 кварталу (II року).

Індекс доходності та середньорічна рентабельність проекту:

Індекс доходності (ІД) - це відношення сумарного дисконтованого доходу до сумарних дисконтованих витрат.

Повинно дотримуватись співвідношення  $ІД > 1$ . Оскільки  $1.4294 > 1$ , то по цьому показнику проект можна рекомендувати до впровадження. Тоді середньорічна рентабельність проекту ( $R$ ) буде:

$$R = \frac{ІД}{n} \cdot 100\% = \frac{1.4294}{8} \cdot 100\% = 17.8\%$$

Таким чином, даний проект є високорентабельним. За один рік і 40 днів компанія зможе розрахуватись з інвестором, а наступний залишок року буде працювати на власний прибуток. Термін окупності даного проекту визначається

періодом між першим та другим роками (коли  $ЧПВ > 0$ , то проект окупається).

### **Висновок до розділу**

В ході технічного аудиту та аналізу ринкових можливостей запуску стартап-проекту, було визначено що ПЗ має переваги в кількості та якості функціоналу в порівнянні з іншими, вже присутніми на ринку ПЗ. Також ціна є нижчою за середню ринкову, що також збільшує ймовірність успіху. Також наше ПЗ має унікальне позиціонування. До слабких сторін належать: репутація бренду та частка ринку. Ця проблема вирішується зі збільшенням тривалості перебування на ринку.

Конкуренція на ринку є високою та клієнти не виявляють бажання змінювати поточне ПЗ, тому для цього було розроблено ряд альтернатив ринкового впровадження, найкращими з яких є надання безкоштовного тестового періоду та програма лояльності при придбанні більш ніж 6 місячної підписки. Отже, є дуже висока ймовірність успішної ринкової комерціалізації проекту, адже існує попит на ПЗ даного типу, динаміка ринку є позитивною, рентабельність роботи на ринку є високою 78%.

Проаналізувавши ринок ми маємо гарні перспективи впровадження з огляду на потенційні групи клієнтів, низькі бар'єри входження та висока конкурентоспроможність проекту.

## ЗАГАЛЬНІ ВИСНОВКИ

У першому розділі було зроблено огляд досліджень в області складання розкладів, розглянуто зміст та основні принципи планування діяльності на підприємстві. Було визначено основні типи планування. Також детально розглянуто оперативно – календарне планування, визначено його основні вимоги, завдання та вхідні данні. Були наведені системи оперативно – календарного планування: календарне планування, диспетчерське регулювання, міжцехове планування, внутрішньоцехове планування. Описано особливості оперативно-календарного планування на підприємствах різних типів виробництва: одиночне, серійне, масове.

Було класифіковано задачі теорії розкладів та визначено основні методи вирішення задач теорії розкладів, до них відносяться: метод динамічного програмування, методи теорії графів, метод гілок та меж, евристичні та ймовірнісні методи. Зроблено огляд досліджень за темою енергетично ефективних задач та сформовано змістовну постановку задачі.

У другому розділі була наведена математична постановка задачі. Були досліджені властивості задачі та позначення, що використовуються в алгоритмі. Також було наведено характеристики робіт та директивних термінів. Було наведено алгоритм розподілу робіт між машинами та алгоритм енергетичного розміщення робіт. Послідовно використовую даний алгоритм ми можемо скласти енергетично ефективний розклад та розрахувати кількість енергії, що буде використана. Також було наведено приклад, що демонструє роботу алгоритму. Алгоритм надає рішення, що мінімізує загальну кількість використаної енергії.

У третьому розділі були проведені експериментальні дослідження. У ході проведення експериментів було визначено, що розроблений алгоритм має більшу ефективність в порівнянні з алгоритмом для паралельних машин з визначеним часом надходження та директивними термінами. Було виділено 27 класів задач з різними вхідними параметрами, середньою тривалістю робіт, дисперсією тривалості та



дисперсією директивних термінів. Визначено погіршення роботи алгоритму при збільшенні дисперсії тривалості робіт.

У четвертому розділі були наведені засоби розробки програмного забезпечення.

Була визначена архітектура та описані клієнтські та серверні засоби розробки. Також наведена схема структурна пакетів, що містить 4 пакети. На схемі структурній компонентів відображений взаємозв'язок між 12 компонентами програмного забезпечення.

У п'ятому розділі в ході технічного аудиту та аналізу ринкових можливостей запуску стартап-проекту, було визначено що ПЗ має переваги в кількості та якості функціоналу в порівнянні з іншими, вже присутніми на ринку ПЗ. Також ціна є нижчою за середню ринкову, що також збільшує ймовірність успіху. Також наше ПЗ має унікальне позиціонування. До слабких сторін належать: репутація бренду та частка ринку. Ця проблема вирішується зі збільшенням тривалості перебування на ринку.

Конкуренція на ринку є високою та клієнти не виявляють бажання змінювати поточне ПЗ, тому для цього було розроблено ряд альтернатив ринкового впровадження, найкращими з яких є надання безкоштовного тестового періоду та програма лояльності при придбанні більш ніж 6 місячної підписки. Отже, є дуже висока ймовірність успішної ринкової комерціалізації проекту, адже існує попит на ПЗ даного типу, динаміка ринку є позитивною, рентабельність роботи на ринку є високою 78%.

Проаналізувавши ринок ми маємо гарні перспективи впровадження з огляду на потенційні групи клієнтів, низькі бар'єри входження та висока конкурентоспроможність проекту.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Д.О. Волошин, В.М. Клименко. О.Г.Жданова, М.О. Сперкач, О.А. Халус, “Дослідження задачі визначення максимально пізнього моменту початку виконання робіт з мінімізацією сумарного випередження відносно директивних термінів виконання робіт”, МОДС 2019 Тези доповідей, с. 398-401.
2. В.М. Клименко. О.Г.Жданова, М.О. Сперкач, О.А. Халус, “Дослідження задачі визначення енергетично ефективних розкладів для паралельних машин”, Третя всеукраїнська науково-практична конференція молодих вчених та студентів «Інформаційні системи та технології управління», Київ, с.97-101, 2019.
3. В.М. Клименко. О.Г.Жданова, М.О. Сперкач “Складання енергетично ефективних календарних планів для функціонування виробничих систем”, Журнал “Вісник Вінницького політехнічного інституту” стаття.
4. Зміст та основні принципи планування діяльності підприємства [Електронний ресурс] – Режим доступу до ресурсу: [https://pidruchniki.com/84332/ekonomika/planuvannya\\_diyalnosti\\_pidpriyemstva](https://pidruchniki.com/84332/ekonomika/planuvannya_diyalnosti_pidpriyemstva).
5. Зміст і завдання оперативно-календарного планування [Електронний ресурс] – Режим доступу до ресурсу: <http://www.readbook.com.ua/book/31/800/>.
6. Оперативно-календарне планування на підприємстві [Електронний ресурс] – Режим доступу до ресурсу: <http://www.d-learn.pu.if.ua/data/users/7588/import/%D0%A2%D0%B5%D0%BC%D0%B0%206%20%D0%9F%D0%94%D0%9F.pdf>.
7. Планово-облікові одиниці та сфера їх застосування [Електронний ресурс] – Режим доступу до ресурсу: [https://studopedia.su/2\\_41592\\_planovo-oblikovi-odinitsi-ta-sfera-ih-zastosuvannya.html](https://studopedia.su/2_41592_planovo-oblikovi-odinitsi-ta-sfera-ih-zastosuvannya.html).
8. Классификация задач теории расписаний [Електронний ресурс] – Режим доступу до ресурсу: <https://studfiles.net/preview/6050916/page:2/>.

9. ТЕОРИЯ РАСПИСАНИЙ ЗАДАЧИ И АЛГОРИТМЫ [Электронный ресурс] – Режим доступа до ресурсу: <http://physcontrol.phys.msu.ru/materials/PosobieLazarev/TeorRasp.pdf>.
10. Р. Беллман Динамическое программирование М.: ИЛ, 1960. 400 с
11. Теорія графів [Электронный ресурс] – Режим доступа до ресурсу: [https://uk.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D1%96%D1%8F\\_%D0%B3%D1%80%D0%B0%D1%84%D1%96%D0%B2](https://uk.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D1%96%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D1%96%D0%B2).
12. Обробка інформації в складних організаційних системах [Электронный ресурс] – Режим доступа до ресурсу: [http://www.hups.mil.gov.ua/periodic-app/article/11427/soi\\_2014\\_2\\_20.pdf](http://www.hups.mil.gov.ua/periodic-app/article/11427/soi_2014_2_20.pdf).
13. Алгоритм для решения задачи коммивояжера / Д.Ж. Литтл, К. Мурти, Д. Суини, К. Кэрел // Экономика и математические методы. – 1965. – Т. 1, вып. 1. – С. 90-107.
14. Бурков В.Н. Элементы теории графов / В.Н. Бурков, Д.А. Новиков. – 2001. – С. 28.
15. Танаев В.С., Шкурба В.В. Введение в теорию расписаний. — М.: Наука, 1975. — 256 с.
16. Методи розв’язання задач теорії розкладів [Электронный ресурс] – Режим доступа до ресурсу: <https://dspace.uzhnu.edu.ua/jspui/bitstream/lib/4685/2/%D0%A0%D0%9E%D0%97%D0%941~1%285-23%29.DOC>.
17. F. Yao, A. Demers, S. A . Shenker, “Scheduling model for reduced CPU energy,” in 36th Annual Symposium on Foundation of Computer Science (FOCS 1995) Milwaukee, Wisconsin, 1995, pp. 374–382.
18. А. С. Аничкин, В. А. Семенов, “Современные модели и методы теории расписаний,” Труды ИСП РАН, том 26, №3, с. 5-50. 2014.

19. Ю.О. Верес, “Розподіл обмежених ресурсів в управлінні проектами,” Вісник Національного університету "Львівська політехніка", № 685, с. 33-44. 2010.
20. А.В. Кононов, “Актуальные задачи теории расписаний: вычислительная сложность и приближенные алгоритмы.” дис. канд. физ-мат. наук., федер. гос. учрежд. инс. математики им. С.Л. Соболева, Новосибирск, 2014.
21. И.Н. Лушакова, “Задачи теории расписаний для системы с нефиксированными маршрутами и ресурсными ограничениями.” дис. канд. физ-мат. наук., Белорусский гос. унив., Минск, 1992.
22. Д.И. Архипов, А.А. Лазарев, Г.В. Тарасов, “Определение загрузки ресурсов при поиске нижних оценок для задачи RCPSP,” Прикладная математика и вопросы управления, №3, с. 35-46. 2017.
23. Herroelen W., Demeulemeester E., De Reyck B. A classification scheme for project scheduling // Project Scheduling. Recent Models, Algorithms and Applications. Ch. 1. — Boston, London, Dordrecht: Kluwer Academic Publisher, 1999. — P. 1–26.
24. Гимади Э.Х., Глебов Н.И. Экстремальные задачи принятия решений. — Новосибирск: Изд-во Новосиб. ун-та, 1982.
25. Танаев В.С., Шкурба В.В. Введение в теорию расписаний. — М.: Наука, 1975. — 256 с.
26. Танаев В.С., Гордон В.С., Шафранский Я. М. Теория расписаний. Одностадийные системы. — М.: Наука. Гл. ред. физ.-мат. лит., 1984. — 384 с.
27. Graham R.L., Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G. Optimization and approximation in deterministic scheduling: a survey // Annals of Discrete Mathematics — 1979. — Vol. 5 — P. 287–326.
28. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. — М.: Мир, 1985. — 512 с.
29. PHP [Электронный ресурс] – Режим доступа до ресурсу: <https://ru.wikipedia.org/wiki/PHP>.

30. Java [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/Java>.
31. .NET [Электронный ресурс] – Режим доступа до ресурсу:  
[https://ru.wikipedia.org/wiki/.NET\\_Framework](https://ru.wikipedia.org/wiki/.NET_Framework).
32. Perl [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/Perl>.
33. Python [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/Python>.
34. Ruby [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/Ruby>.
35. Хостинг – провайдер [Электронный ресурс] – Режим доступа до ресурсу:  
[https://uk.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3%D0%BE%D0%B2%D0%B0\\_%D0%BA%D0%BE%D0%BC%D0%BF%D0%B0%D0%BD%D1%96%D1%8F](https://uk.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3%D0%BE%D0%B2%D0%B0_%D0%BA%D0%BE%D0%BC%D0%BF%D0%B0%D0%BD%D1%96%D1%8F).
36. HTML [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/HTML>.
37. JavaScript [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/JavaScript>;
38. Mysql [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/MySQL>.
39. PostgreSQL [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/PostgreSQL>.
40. SQLite [Электронный ресурс] – Режим доступа до ресурсу:  
<https://www.sqlite.org/index.html>.
41. mSQL [Электронный ресурс] – Режим доступа до ресурсу:  
<https://en.wikipedia.org/wiki/MSQL>.

42. Oracle [Электронный ресурс] – Режим доступа до ресурсу:  
<https://habr.com/ru/hub/oracle/>.
43. dBm [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/DBm>.
44. Hyperware [Электронный ресурс] – Режим доступа до ресурсу:  
<https://www.pcmag.com/encyclopedia/term/44570/hyperware>.
45. Informix [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/Informix>.
46. InterBase [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/Interbase>.
47. Sybase [Электронный ресурс] – Режим доступа до ресурсу:  
[http://citforum.ck.ua/database/sql\\_any/sql\\_01.shtml](http://citforum.ck.ua/database/sql_any/sql_01.shtml).
48. ECMAScript [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/ECMAScript>.
49. React [Электронный ресурс] – Режим доступа до ресурсу: <https://ru.reactjs.org/>.
50. Angular [Электронный ресурс] – Режим доступа до ресурсу: <https://angular.io/>.
51. Vue.js [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/Vue.js>.
52. Node.js [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/Node.js>.
53. Electron [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/Electron>.
54. NW.js [Электронный ресурс] – Режим доступа до ресурсу:  
<https://spark.ru/startup/nwjs-ide/blog/39778/nw-js-sreda-dlya-razrabotki-desktopnih-prilozhenij>.
55. React Native [Электронный ресурс] – Режим доступа до ресурсу:  
<https://facebook.github.io/react-native/>.

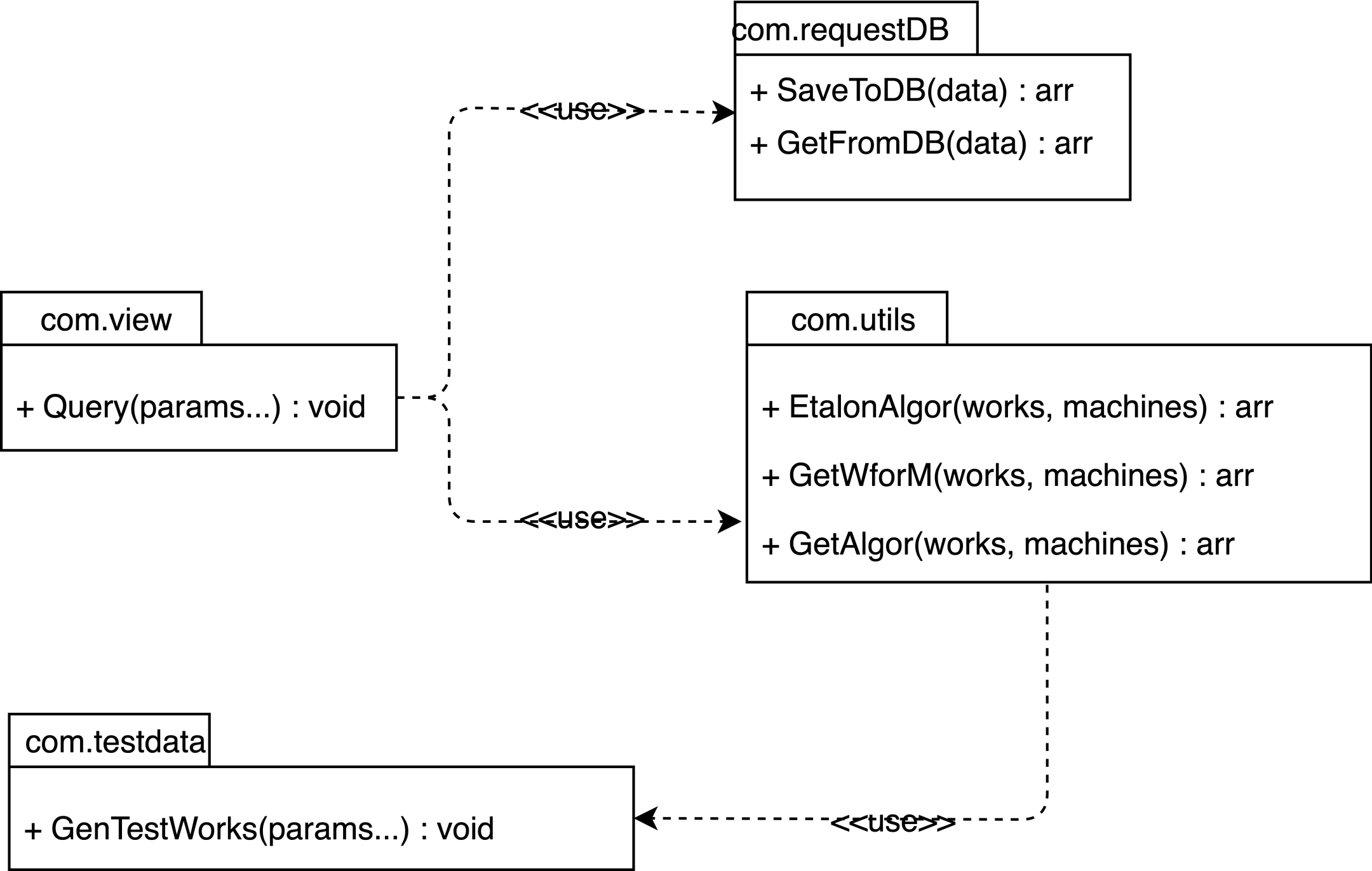
56. Cordova [Электронный ресурс] – Режим доступа до ресурсу:  
<https://cordova.apache.org/docs/ru/latest/guide/overview/>.
57. Adobe Creative Suite [Электронный ресурс] – Режим доступа до ресурсу:  
[https://uk.wikipedia.org/wiki/Adobe\\_Creative\\_Suite](https://uk.wikipedia.org/wiki/Adobe_Creative_Suite).
58. Apache JMeter [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/JMeter>.
59. Ajax [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/AJAX>.
60. DHTML [Электронный ресурс] – Режим доступа до ресурсу:  
<https://htmlweb.ru/html/dhtml.php>.
61. XMLHttpRequest [Электронный ресурс] – Режим доступа до ресурсу:  
<https://developer.mozilla.org/ru/docs/Web/API/XMLHttpRequest>.
62. JSON [Электронный ресурс] – Режим доступа до ресурсу:  
<https://ru.wikipedia.org/wiki/JSON>.
63. Google Maps [Электронный ресурс] – Режим доступа до ресурсу:  
[https://uk.wikipedia.org/wiki/%D0%9A%D0%B0%D1%80%D1%82%D0%B8\\_Google](https://uk.wikipedia.org/wiki/%D0%9A%D0%B0%D1%80%D1%82%D0%B8_Google).
64. Google Suggest [Электронный ресурс] – Режим доступа до ресурсу:  
[https://ru.wikipedia.org/wiki/Google\\_Suggest](https://ru.wikipedia.org/wiki/Google_Suggest).
65. UNIX система [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/UNIX>.
66. PhpStorm [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/PhpStorm>.
67. IntelliJ IDEA [Электронный ресурс] – Режим доступа до ресурсу:  
[https://ru.wikipedia.org/wiki/IntelliJ\\_IDEA](https://ru.wikipedia.org/wiki/IntelliJ_IDEA).
68. WebStorm [Электронный ресурс] – Режим доступа до ресурсу:  
<https://uk.wikipedia.org/wiki/WebStorm>.

69. Symfony2 [Електронний ресурс] – Режим доступу до ресурсу:  
<https://ru.wikipedia.org/wiki/Symfony>.
70. Yii [Електронний ресурс] – Режим доступу до ресурсу:  
<https://yiiframework.com.ua/ru/doc/guide/2/>.
71. UML [Електронний ресурс] – Режим доступу до ресурсу:  
<https://ru.wikipedia.org/wiki/UML>.
72. CSS [Електронний ресурс] – Режим доступу до ресурсу:  
<https://uk.wikipedia.org/wiki/CSS>.
73. Архітектура клієнт-сервер [Електронний ресурс] – Режим доступу до ресурсу:  
[https://en.wikipedia.org/wiki/Client%E2%80%93server\\_model](https://en.wikipedia.org/wiki/Client%E2%80%93server_model).
74. Стартап [Електронний ресурс] – Режим доступу до ресурсу:  
<https://uk.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D1%80%D1%82%D0%B0%D0%BF>.
75. Аналіз п'яти сил Портера [Електронний ресурс] – Режим доступу до ресурсу:  
[https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B0%D0%BB%D1%96%D0%B7\\_%D0%BF%27%D1%8F%D1%82%D0%B8\\_%D1%81%D0%B8%D0%BB\\_%D0%9F%D0%BE%D1%80%D1%82%D0%B5%D1%80%D0%B0](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B0%D0%BB%D1%96%D0%B7_%D0%BF%27%D1%8F%D1%82%D0%B8_%D1%81%D0%B8%D0%BB_%D0%9F%D0%BE%D1%80%D1%82%D0%B5%D1%80%D0%B0).
76. SWOT-аналіз [Електронний ресурс] – Режим доступу до ресурсу:  
<https://uk.wikipedia.org/wiki/SWOT-%D0%B0%D0%BD%D0%B0%D0%BB%>.



**ДОДАТКИ****ДОДАТОК А****Графічні матеріали**

# Плакат 2. Схема структурна пакетів

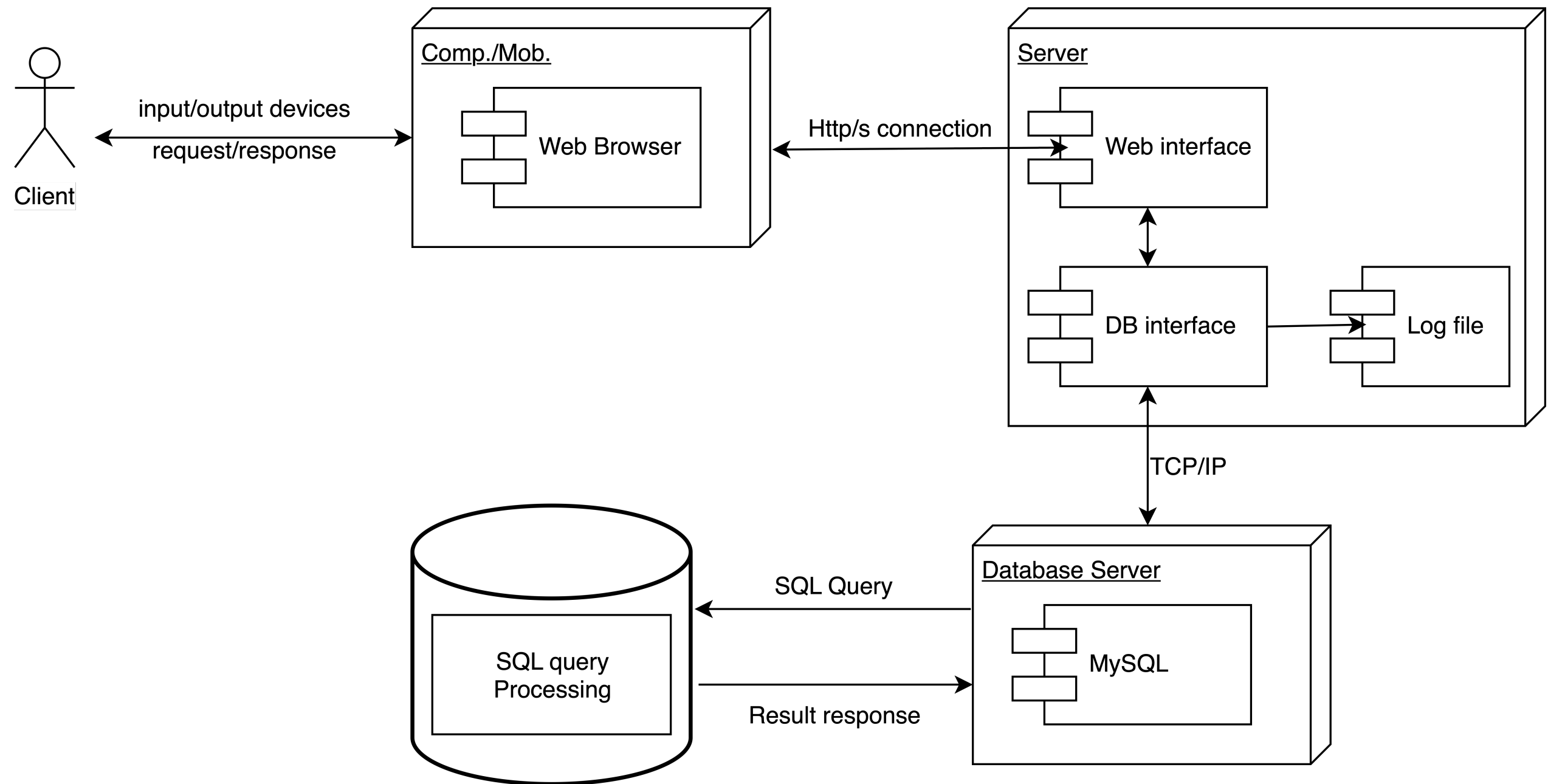


Демонстраційний плакат до магістерської дисертації  
на тему «Складання енергетично ефективних розкладів в оперативно-календарному плануванні»

Виконав студент гр. ІС-81мп  
Керівник

Клименко В.М.  
Сперкач М.О.

# Плакат 1. Схема структурна розгортання



Демонстраційний плакат до магістерської дисертації  
на тему «Складання енергетично ефективних розкладів в оперативно-календарному плануванні»

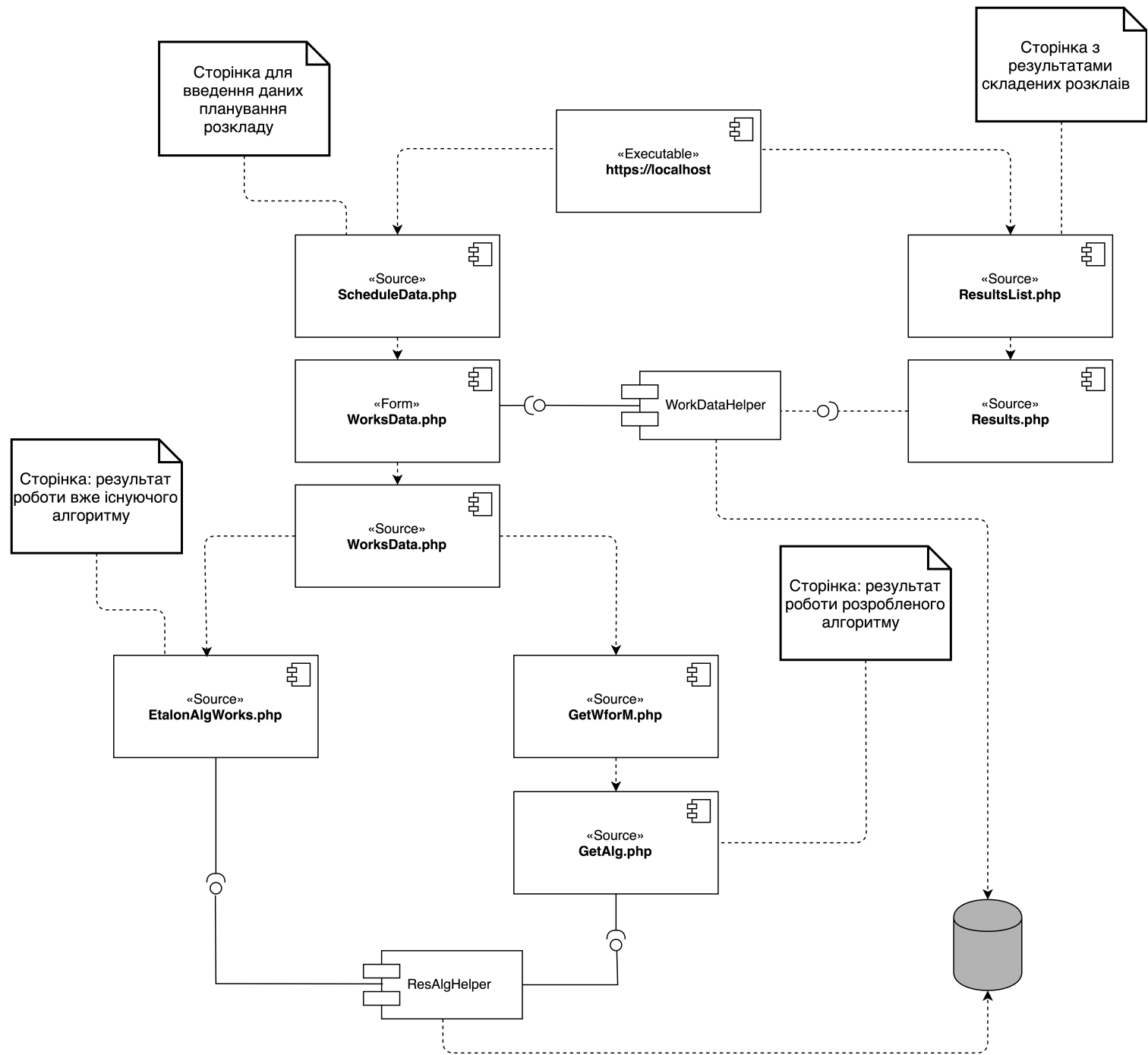
Виконав студент гр. ІС-81мп

Клименко В.М.

Керівник

Сперкач М.О.

# Плакат 3. Схема структурна компонентів

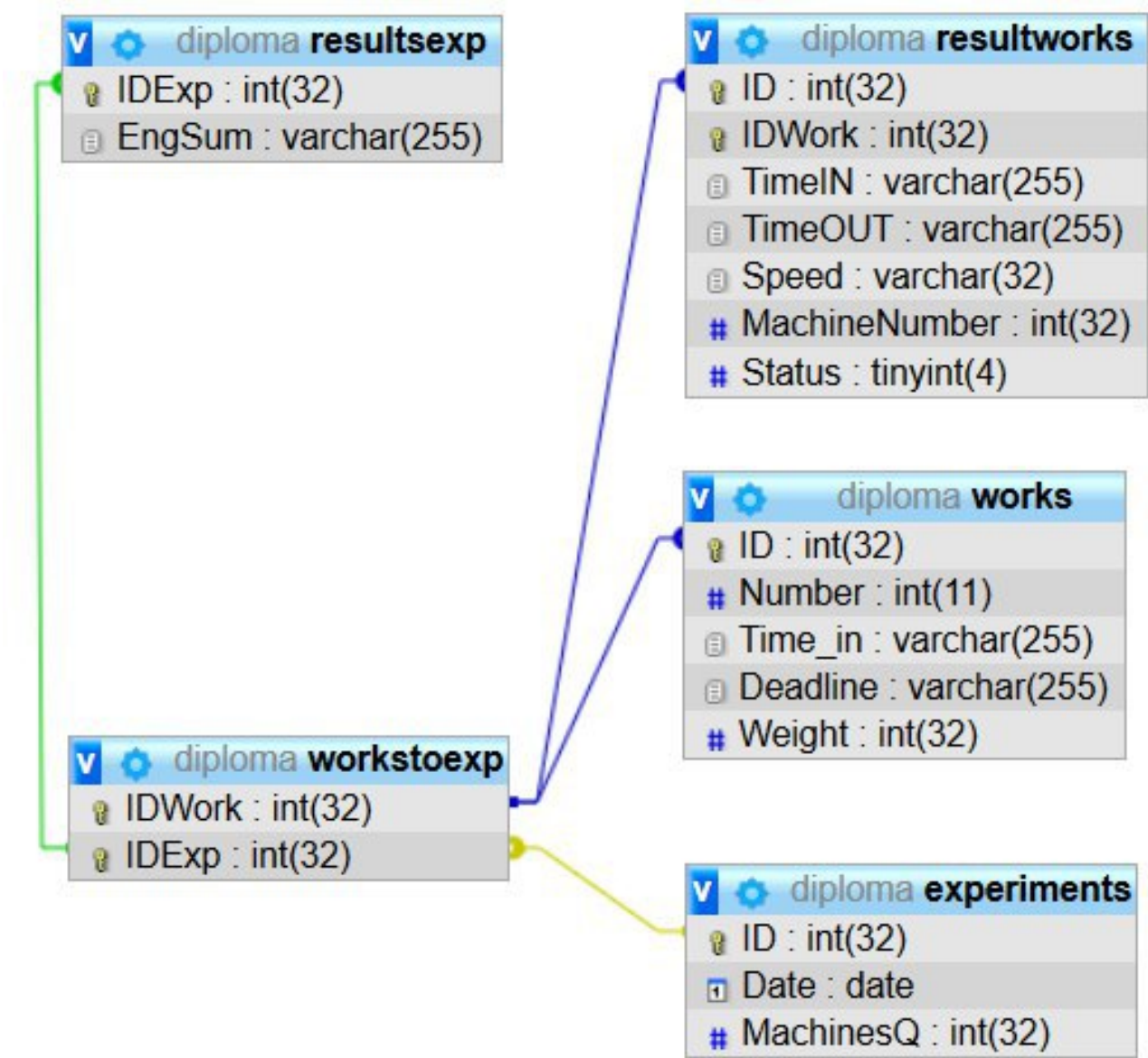


Демонстраційний плакат до магістерської дисертації  
на тему «Складання енергетично ефективних розкладів в оперативно-календарному плануванні»

Виконав студент гр. ІС-81мп  
Керівник

Клименко В.М.  
Сперкач М.О.

# Плакат 4. Схема структурна бази даних



Демонстраційний плакат до магістерської дисертації  
на тему «Складання енергетично ефективних розкладів в оперативно-календарному плануванні»

Виконав студент гр. ІС-81мп

Клименко В.М.

Керівник

Сперкач М.О.